MTP-329

COGENT

DM-1

FORGE

GIS

IDS

MANAGE

MARK IV

NIPS

RAPID

TDMS

Data Management Systems Survey

JANUARY 1969

THE MITRE CORPORATION

DATA MANAGEMENT SERIES-NO. 2

This document has been approved for public release and sale; its distribution is unlimited.

198

MTP-329

DATA MANAGEMENT SYSTEMS SURVEY

JANUARY 1969

J. P. FRY
S. BRAMSON
D. C. FRIED
W. P. GRABOWSKY
J. JEFFRIES, JR.
S. B. MAHLE
H. J. STERNICK

THE MITRE CORPORATION
Washington, D.C. Operations

The work reported in this document was sponsored by the Defense Communications Agency under Contract F19628-68-C-0365

ABSTRACT

This report presents the results of a survey of salient characteristics of a representative set of state-of-the-art data management systems. It is part of an effort to identify the state-of-the-art capabilities of data management systems for third-generation computer systems.

Section I of the report includes general descriptions of the systems surveyed and establishes the terminology for logical organization of data used in the survey.

Section II describes the capabilities surveyed and presents the survey results in tabular format.

FOREWORD

This survey is the second of a projected series of reports relating to the state-of-the-art of data management systems. Others will present a comparative analysis of selected data management systems including detailed analysis of individual systems.

The survey was performed in two phases. Initially, under the aegis of the Defense Communications Agency the survey was performed in the fall of 1967 as part of DCA's technical support to the Joint Technical Specification Group. This group, under the guidance of the Information Systems and Standards Division of the Joint Command and Control Requirements Group, was tasked with the preparation of the specifications for the standard EDP source selection for the fixed headquarters of the World Wide Military Command and Control System (WWMCCS).

Subsequent problems delayed the public release of this report so that the survey had to be validated again in the fall of 1968 due to this rapidly changing technological area.

Some of the surveyed systems are still in the development stage and their capabilities are not documented in the open literature. In order to perform the survey, it was necessary to visit the developers, to discuss the systems in detail with knowledgeable individuals, and to record their responses.

The work was performed under the direction of Mr. J. A. Gosden, Head, Information Systems Subdepartment, with Mr. J. P. Fry as the principal investigator.

The team members included: Messrs. S. Bramson, D. C. Fried, W. P. Grabowsky, J. Jeffries, Jr., H. J. Sternick and Mrs. S. B. Mahle.

TABLE OF CONTENTS

SECTION I

INTRODUCTION TO THE SURVEY

APPROACH

This document presents the results of a survey of selected characteristics of a representative set of state-of-the-art data management systems. It is part of an effort to identify the state-of-the-art capabilities of data management systems for third generation computer systems.

The surveyed systems included in this document are: Auerbach Corporation's DM-1, Burroughs Corporation's FORGE, U. S. Army's RAPID*, Computer Sciences Corporation's COGENT III, General Electric Corporation's IDS, Informatics Incorporated's MARK IV, International Business Machines Corporation's GIS, the National Military Command System Support Center's NIPS, Scientific Data Systems Corporation's 9 Series MANAGE, and System Development Corporation's TDMS.

In order to prepare this survey, three major difficulties had to be surmounted:

(a) lack of precise and universal terminology

(b) inconsistent interpretation of general capabilities, and

(c) inadequate documentation.

There is no standard universally accepted set of terminology. Various systems use different terminology to express the same concepts, and not even the more commonly used terms describe data management system concepts unequivocally. To alleviate this problem, standard terminology was adopted and used throughout the entire survey. This terminology is defined below.

* Developed by Control Data Corporation

One important consideration dictated the entire approach to conducting the survey. This was the necessity to examine each system in a consistent manner. Significant general data management system capabilities and data on availability of the systems were compiled and arranged into eight categories: file definition and organization, data retrieval, maintenance, output presentation, data access methods, general interface capability, system statistics, and system availability. Each state-of-the-art data management system was examined in terms of these categories.

Members of the survey staff studied all available system documentation and interviewed representatives of all developers. Because of varying system design philosophies and technical backgrounds of representatives, each developer responded in an individual manner. The individual questions on the survey form were discussed by the staff interviewers and developers to clarify the questions and assure consistent responses.

There are several limitations to the survey. Not all capabilities are applicable to all systems because of the wide variation in system philosophies. In some instances, developers refused to divulge any information at all because their data management system design is in an early stage.

The reader of this survey should understand that the survey is not an evaluation. No attempt has been made to weight the capabilities, that is, to indicate which ones are more significant. The order in which they occur or the detail of their description in no way reflects their overall importance, and the list of capabilities itself may not be exhaustive. It should also be noted that certain capabilities have converse relationships; that is a "yes" answer for one automatically implies a "no" for another. It must also be remembered that not all of the capabilities may be desirable for a particular user's requirements. No measurement of the ease or difficulty of using the systems has been attempted, nor has any attempt been made to estimate response times. System evaluation is, at best, extremely complex and time consuming and this document is in no way intended to be an evaluation of state-of-the-art data management systems.

The time frame chosen for the survey was that the initial version of the data management systems surveyed must be demonstrable by the fourth quarter of calendar year 1968. Some of the systems, FORGE, IDS, MARK IV, and 9 Series MANAGE, have already achieved this level of capability.

The information supplied to us by the developers reflects the capabilities of their systems as defined at the time the survey was conducted. The information supplied by the developers in no way commits or restricts them to capabilities enumerated in the survey.

SYSTEMS OVERVIEWS

Auerbach/DM-1

Data Manager-1 (DM-1) is a generalized data management software system designed by Auerbach Corporation as a proprietary package. The system is being implemented on two distinctly different hardware systems: on a UNIVAC 1218 (Mil Spec 418) for the U. S. Air Force under contract through the Rome Air Development Center; and on the IBM 360/50 for the Western Electric Corporation. The system will generally be available as a proprietary package.

The DM-1 system provides program and job library services, data storage and access services, and job exec tion control. The job library contains a number of general-purpose system jobs for building, maintaining, and querying the data base and job library. User jobs can be added to this library. The structure of the data pool provides logically for definition of items and defines a structure for a set of directory tables.

Basically DM-1 provides a nucleus system with emphasis on data definition, data structuring, and data access methods. Features such as generalized report generators, user query and maintenance languages, and recording services are not defined in the DM-1 design. It is intended that these items be developed for each customer as separate units to be integrated into the system. Programs (generalized report generators) would be added to the job library with no unique status. The syntax of system languages (query maintenance) can be defined by the user through a meta-language (similar to ALGOL).

DM-1 has been designed independently of a specific computer. However, the system is designed to utilize fully the capabilities of whatever operating system that DM-1 is to run under. A standard interface is provided to the computer through the use of the data service routines, which are theoretically the only computer dependent routines of the system.

DM-1 has a very comprehensive logical file structuring capability. The system has diverged from the narrow philosophy of one-level serial formatted files to permit the complex hierarchies of logical relationships in the form of general linked tree structures. The data description language permits the use of variable-length data elements, optional data elements, and nested structures.

The DM-1 data pool is a series of fixed-length data segments containing an unformatted stream of bits. The segment size of the data pool is independent of the logical file organization and theoretically utilizes the maximum buffer size of the operating system. The DM-1 access mechanism, which utilizes the IOCS, transfers the data between the data pool and core. The segmented data stream is interpreted in core by the data service routines with the aid of the system directories and the segment index.

Burroughs/FORGE

File Organization Generator (FORGE) is a programming system designed and developed by the Burroughs Corporation for users of the B5000 series of equipments. FORGE first became available in 1967 for the B5500 computer.

FORGE, which is written in the COBOL language, is designed through the use of table look-up techniques, to facilitate the organization, storage, maintenance and retrieval of information. FORGE operates under the Burroughs Master Control Program (MCP) and uses the Burroughs "head-per-track" disc file for information storage.

The system provides for automatic generation of file-building and/or file-maintenance programs; automatic generation of search routines for each file to facilitate retrieval of any record or records from that file; updating of multiple files from single transactions; transaction processing from various input devices; preparation of printed reports on results of maintenance transactions; processing of user-defined transactions, user-inserted COBOL procedures and subroutines; and production of user-defined reports.

FORGE generates a COBOL-language source program from information provided by the user in a set of parameter cards. The parameter cards describe type of input and output media and the structure and content of all disc data files.

FORGE provides the same logical and physical data structure as that available in COBOL. Data items within each logical record of a disc file are stored contiguously, while logical records are stored randomly.

FORGE does not provide a formal retrieval language or a formal output formatting program. However, the user can include any COBOL procedures that are available in the Burroughs COBOL system via special parameter cards. Using this facility, the user can provide the special procedures he desires in processing the FORGE-generated files. Thus, theoretically, a user can write his own retrieval language and/or an output-formatting program and include it with his FORGE parameter cards.

Computer Sciences Corporation/COGENT III

COGENT III, a COBOL-compatible generalized file management system, is an advanced general-purpose data management system developed by Computer Sciences Corporation (CSC) for the IBM/360 family of equipment. COGENT III represents the next development in a series of generalized file management systems produced by CSC. It follows COGENT II, also developed for the IBM/360 family, and a series of COGENT systems developed for the IBM 7090, the UNIVAC 1107/1108, the IBM 7044, and the RCA SPECTRA 70.

COGENT III is modular in construction and is designed to function on a wide range of System/360 configurations. It provides for use of unit-record, magnetic-tape and direct-access devices as well as of teleprocessing equipment. The system employs and is compatible with O/S 360 and the related COBOL compiler.

COGENT III generalizes the major data processing functions associated with information storage, maintenance, retrieval, and presentation. The system allows for use of a common data base by various groups of users. This facility is provided under the control of a single, comprehensive data directory which describes all data used or referenced by the system as well as the relationships between data sets, records, and data fields in the data base.

COGENT III consists of an information-system language and a language processor (impler ented in COBOL) which interprets this language and generates functional COBOL programs for maximum machine-independence to perform the information system tasks requested. In addition, an interactive inquiry retrieval and storage language is provided to meet rapid response requirements. This language is processed by the COGENT III Interpretive Processor, which is implemented in assembly language employing re-entrant code. The system provides for specification and generation of program(s) for a simple function, an entire data processing application, and/or a series of integrated applications as well as for interpretive processing of dynamic requests for information storage and retrieval, including on-line data input.

The information-system language consists of fixed-form data descriptions and functional specifications as well as an interactive inquiry and storage language. The language elements are designed to allow specification of the information needs of information collection, storage, maintenance, retrieval, and presentation for various levels of users--management, operating personnel and system analysts.

In general the language allows the user to specify the following:

(a) Information to be stored in the data base

(b) Source of the information

(c) How and under what conditions the information is created and maintained

(d) What relationships exist between different collections of information (hierarchical structure and integration between data sets)

(e) Who can have access to modify or retrieve the information

(f) How and under what conditions the information is to be retrieved

(g) How the information is to be presented

(h) On-line, off-line, batched, real-time, or demand processing of tasks

The data descriptions allow the user to describe such items as data fields, relationships between data fields, security codes for maintenance and retrieval, decoding and encoding association tables, physical characteristics of data elements, presentation format of data elements, which fields may be used as access keys for content retrieval, and which fields are record identifiers.

The functional specifications allow the user to specify the task control information file, creation file, maintenance, output sort, and report format, the processing mode (direct, demand, or batch), and the file structure (serial or direct).

The interactive inquiry retrieval and storage language allows the user to access the data base via "ad hoc" queries and execution of previously defined queries. The user can obtain immediate response or can batch queries.

Hierarchical data structures, which can extend over more than one data set, are provided for both sequential and direct-access data sets. In most cases, hierarchical records physically follow their associated higher-level records. In the case of multiple data sets, different levels of hierarchical records may be physically stored on different devices (direct access only).

When stored on direct-access devices, the record keys as defined by the data description language are employed to build a record key index. Records are indexed by hierarchical group only. Individual records within the hierarchy are extracted in core from physical blocks containing a hierarchical group.

An access key index is automatically constructed for each data field defined as an access field in the data description language. The index contains an entry for each possible field value with pointers (record keys) to all physical blocks which contain the value. These indexes are employed to allow content retrieval.

Control Data Corporation/RAPID

Random Access Personnel Information Disseminator (RAPID) is a specialized data management system in the process of development by the Control Data Corporation for the United States Army under a contract awarded in July 1967. The system will be used to maintain and retrieve data on U. S. Army officers, enlisted and civilian personnel, and military strength. RAPID will operate on the Control Data 3300 under Multi Access Shared Time Execution Routine (MASTER). Personnel managers will communicate with the system via on-line remote terminals.

The user can define the logical structure of the file he is using, the format of the input data, and the security codes of his file; and he can specify the fields he wants to use as indices. Serial searching is performed on non-indexed fields.

RAPID accepts transactions from the field and processes them through subroutines which convert them to a common internal format. After validation, RAPID updates both the data base and the indices.

The retrieval language of RAPID is designed to permit the user a large degree of flexibility. The user enters conditional expressions describing a record or file subset, data-reduction requests including the ability to create a matrix, and various report formats.

RAPID retains retrieval data in a summary file until it is needed, saves jobs for repetitive executions, calls out standard jobs for processing upon request, monitors security access to the files, and performs various other control functions.

Maintenance for the RAPID system is a special-purpose function. Transactions are originated and coded in the field and transmitted for central processing. COBOL programs validate the data and convert the accepted transactions to a standard internal format. RAPID then accepts the reformatted transactions and automatically updates the data files and indices. An audit trail and a listing of valid transactions is produced as a by-product of the maintenance run. Due to the nature of the user-defined problem specifications, RAPID does not have a general-purpose maintenance capability in the first version.

General Electric/IDS

Integrated Data Store (IDS) is a data management system designed and developed by the General Electric Company in 1965. It has been implemented on the GE200, 400, and 600 series of computers.

IDS, in concert with COBOL, provides the capability for generating, maintaining, and retrieving data from disc-resident files. IDS operates under the control of the GE multi-programming monitor which provides an on-line remote terminal inquiry capability.

IDS is embedded in COBOL. All the features available in COBOL are available to the IDS user. Some of the COBOL procedures statements have been modified, and other statements have been added to the basic language in order to accommodate IDS structured files. All maintenance, retrieval, and outputting is accomplished by using COBOL. Theoretically, a distinct set of procedures, a unique COBOL program, might be required for each type of update or query.

IDS provides the facility for placing data in the file and retrieving from it. The functions performed by IDS itself can be likened to that of special input and output subroutines.

A variety of physical and logical file structures are possible under IDS. The user has the traditional capability of specifying the logical structure of his data; he also has some control over the physical structure. IDS provides the capability of linking together any combinations of master or repeating groups to form an entry. During file maintenance, linkages are changed to reflect any additions or deletions to the file. Neither groups nor entries need be physically stored sequentially. They may be stored in a random order, since the links connect all the groups constituting an entry and an entry is randomized on its key element(s).

IDS on the GE400 and 600 series of computers operates under the multi-programming monitor and functions like any other processor (e.g., FORTRAN) in the environment.

There are other basic differences in the versions of IDS available on the GE200, 400, and 600 series. The 200 series does not use COBOL, but an assembly language. The differences between the 400 and 600 series are less pronounced, but they do exist primarily in the COBOL language modifications and additions.

Informatics/MARK IV

The MARK IV file management system is an advanced general-purpose software system developed for the IBM/360 series of equipment. It is the fifth in a series of data management systems designed and developed by Informatics. MARK IV operates under either the Standard Disc Operating System (DOS) or Operating System (OS). The initial system will be available during the latter part of 1967 and a final version will be available early in 1968.

MARK IV is a data processing system designed primarily for business or file applications. The system provides for the generation, maintenance and retrieval of information from tape or disc oriented files. A comprehensive report facility is also provided.

The system generates computer programs based upon specifications given to it by the user. The user employs one or more structured forms to prepare the specifications which become the MARK IV source input. No on-line facilities are now provided by the system.

MARK IV generates a program to perform the user-specified functions from pre-coded routines which are stored in the MARK IV library. The pre-coded routines provide for many of the common data processing tasks, thus reducing the amount of code that has to be generated.

The user specifies the logical organization of his files and the data are stored on tapes or on disc. The data are accessed via the Sequential Access Method (SAM) if the storage medium is tapes, and by the SAM or the Index Sequential Access Method (ISAM) if the storage medium is disc.

The system can: create files from various data sources; maintain files by performing changes, additions and deletions; select data records based on selected criteria; make computations on the data in these records; extract data items from selected records; and produce new files, parts of files, and combinations of files. Various report formats and data arrangements, as well as the facility to use preprinted forms, are available.

In addition to the standard set of programs provided by MARK IV, the user also is provided with the facility for calling up user-coded routines which may be required for special processing. The system also has the facility for saving a generated program in its library, and calling upon the program if the user results are required on a periodic basis.

International Business Machines/GIS

The Generalized Information System (GIS) is a collection of programs written for System 360 to support formatted file retrieval, maintenance, and report-presentation functions. It is a third-generation system which resembles its predecessor, the Formatted File System (FFS). Overall, the system does not appear radically different from its design and implementation basis 7090 FFS (438L).

GIS is a stand-alone system, under control of Operating System 360. Basically, the system operates as a series of job steps in the sequential task environment of Operating System 360 and will be released in two versions. The first release, GIS (Basic), will provide file-handling capabilities which will operate in a non-multiprogrammed environment using batched transactions. A majority of the full procedural language capabilities will be provided in GIS (Basic). The second version will provide additional capabilities, including procedure specification and data entry from local or remote terminals, and operation in a multiprogrammed environment.

GIS is planned for release as an IBM Type II program and, as such, will be produced and maintained by IBM. The first version will require, as a minimum, a 360/40 with a 128K-byte core. Later versions operating in a multiprogrammed environment with teleprocessing capabilities are expected to require a larger core size.

GIS is designed to provide the capabilities of file creation and maintenance, file inquiry, and report preparation in a timely manner. To meet this design objective, GIS provides the user a number of generalized processing modules and an information language through which both data files and file processing procedures can be described. GIS also provides control and utility features to aid the user in performing task specification maintenance, in providing restricted access to files, in providing audit data as a result of file maintenance actions, and in providing detection and recording of specification, data content and system reaction errors.

GIS is built upon the experience gained in its second-generation predecessor, the Formatted File System. Perhaps the most significant feature of GIS is not that it provides the basic necessary capabilities of a generalized file-handling system, but rather the degree of comprehensiveness of these capabilities.

GIS provides additional capabilities over and above the second-generation formatted file features. One of these features is the multi-file capability. The procedural language in GIS allows for up to six files to be referenced during any sub-procedure (QUERY, MODIFY, UPDATE, CREATE, etc.).

Another capability found in GIS that was not generally available in second-generation systems is the ability to handle hierarchical files. GIS can handle up to 15 levels of hierarchy with a single nested group at each level. It also handles multiple types of groups at any level of subordination, so long as these multiple types are not themselves superior to further subordination.

National Military Command System Support Center/NIPS

The NMCS Information Processing System (NIPS/360) is in essence the Formatted File System, FFS, converted to the IBM 360/50 computer system and operational under Operating System 360/Option 2. Historically, the Formatted File Systems began with the SAC 438L system on the IBM 7090 in 1961. The NIPS predecessor on the IBM 1410 was created in 1963 to satisfy the mission of a Naval Fleet Intelligence Center. Since then, the system has been employed at several intelligence centers and commands throughout the world. Basically, the system has evolved in two separate communities: Command and Control, and Intelligence. The Intelligence system, the IDHS FFS, has emphasized efficient processing capabilities and improved output options for its many-volumed data sources. On the other hand, the Command and Control system, NIPS, has expanded logical file maintenance, improved the query language, and built an on-line retrieval mechanism. Both the second-generation FFS systems operate under the modified 1410 operating system OPSYS (1410-PR-155).

The 1410 FFS is being converted to an S/360 model 50H (eight magnetic tape drives, two 2311 discs, one 2302 disc, 2260/1053, and associated card/read/punch printer equipment) in two primary phases. Phase I components include: file structuring, file maintenance, retrieval and sort, output, remote inquiry processing, system-formatted output, and necessary utilities. Included in Phase II are an expanded file maintenance including multi-file, file revision, on-line update, as well as additional components and utilities. The target date for the operational capabilities is January 1968, for Phase I and June 1968, for Phase II.

NIPS/360 is a general-purpose file-handling system operating under O/S 360 and performing the traditional functions of structuring, maintaining, revising and retrieving from a set of data files. While maintaining a high degree of external compatibility with its 1410 predecessor, the internal processing methods have been modified to exploit the capabilities of the third-generation hardware S/360, and operating system O/S 360. Adhering to the strict O/S 360 programming conventions for communication, base register usage, linkage, etc., the conversion is being done using the COBOL language down to the subroutine level of software. At the subroutine level, either COBOL or assembly language will be used depending on which is necessary to efficiently utilize O/S 360 software and S/360 hardware capabilities.

Compatibility with the existing 1410 FFS system was a major design criterion which is being achieved through the ability to use existing FFS control cards, query and summary decks, file maintenance decks, and analyst's procedures.

The new system does much to alleviate the physical restrictions of the old system. Specifically, the physical size restriction of a logical record to 2701 characters has been relieved by making each individual group instance a physical record. However, because of compatibility requirements, some of the logical shortcomings are still apparent in the system.

In the new system, the File Format Table (FFT is the data description or dictionary) will become an integral part of the physical file being described. Thus the NIPS files will be "self-describing" and no longer have a physically remote file description which can easily be misplaced or lost. The S/360 NIPS is primarily a direct-access-oriented system utilizing the capabilities of the disc through the Index Sequential Access Method of O/S 360.

Scientific Data Systems/9 Series MANAGE

SDS 9 Series MANAGE, hereafter referred to as MANAGE, was developed by Scientific Data Systems Corporation (SDS) for the 910, 920, 925, 930, and 9300 computers. The initial system was completed in April 1966 and the final version in January 1967.

MANAGE is a generalized system that provides the capability for generating, maintaining and retrieving data from tape files. Although SDS has been stressing business applications, MANAGE is, in fact, a file-oriented system similar in concept to the MARK III system of Informatics.

The system was designed for and is limited to processing tape files with fixed-length records. Record length may vary between files, but not within a file.

MANAGE operates from the MONITOR 9300 systems tape on the SDS 9300 computer. On the SDS 900 series it operates from a stand-alone MANAGE system tape.

MANAGE has a limited capability to process textual type information such as document abstracts, job status reports and other similar information. The text material must be entered in the file in the same format that is to be printed. Retrieval is accomplished by fixed-position unique identification codes. The capability is further limited by the maximum record size, 15 lines per record and 120 characters per line.

System Development Corporation/TDMS

The Time-Shared Data Management System (TDMS) is a general-purpose system for managing data in a time-sharing environment. It is currently being developed by the Technology Directorate of the System Development Corporation (SDC) as a result of work sponsored in part by the Advanced Research Projects Agency (ARPA) of the Department of Defense. TDMS is being designed and implemented for use on the IBM System 360 Model 50. TDMS is an outgrowth of TSS-Lucid System, also developed by SDC for the AN/FSQ-32 computer.

TDMS will permit the user to describe and generate a file as well as retrieve and display data from the file on a cathode-ray tube (CRT) device. It also will provide the capabilities to update and maintain the file and to generate hard copy reports. TDMS operates under the SDC-provided ADEPT operating system.

In TDMS a file is stored in a series of associated tables. One group of tables holds the actual data element values. There is a separate table for each defined data element, and only unique data element values are stored in these tables (i. e., no duplicate values). Another table contains the names of the data elements in the files. A third type of table describes the individual data elements and their logical relationships. It also points to tables which contain pointers arranged in sorted order, which point to the tables containing the actual data element values.

There are two capabilities in TDMS which can be used to retrieve data. One of these, QUERY, can produce only relatively simple outputs. The second, COMPOSE/PRODUCE, can produce rather sophisticated outputs. Within COMPOSE, the user describes any number of report formats to TDMS. Each report format has a name and requires several statements to describe the data which is to be output, and the way the output will look when it has been produced as a report. PRODUCE provides the user with the capability of requesting any of the report formats previously generated in COMPOSE to produce actual reports.

Through two additional programs, TDMS has capabilities for modifying data element values and for maintaining files. The UPDATE program allows the user to add, delete or change data element values. The MAINTAIN program provides for merging, subsetting, extracting, ordering, and restructuring of files.

DATA STRUCTURE TERMINOLOGY

One of the most difficult problems in describing data management systems is terminology. There is no generally accepted set of terms or definitions. This section defines the terms used in this document to describe the logical organization of data.

There are five basic terms used in this report to represent various levels of aggregation of data. In a scale going from lowest to highest, they are:

Data Element
Group
Logical File
File
Data Base

Data Element

A data element is a single data entity and is a terminal item; that is, it contains no logical substructure and is therefore the lowest level. The most important aspect of a data element is that it is the link between logical and physical structure. In particular, data elements are the only logical entities that have data value(s) which are physically stored. A data element is referenced by an alphanumeric string of characters--its name. The size of a data element may be fixed or variable and is measured by a number of bits, bytes, or characters.

Other terms used for this concept are field, item, atom, element, or property.

Example: Data Element--POPULATION
Data Element Value--202, 779

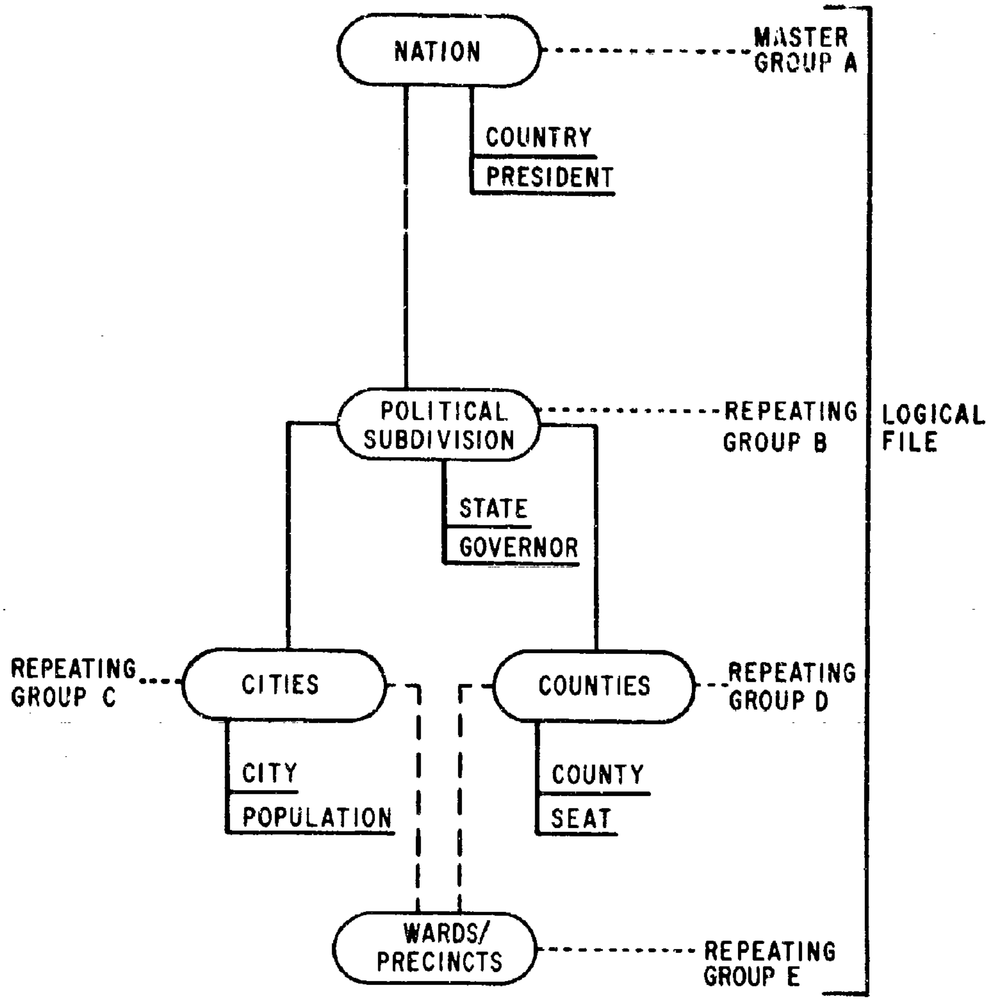
Group

A group is a set of data elements which has an identifier. A group represents a node in the logical hierarchy of data. Groups may vary in frequency, format, and their relationship to each other. These relationships include: subordinate, peer, and superior with respect to other groups. A particular set of data element values for all the data elements comprising the group is called an instance of the group.

Other terms used for this concept are segment, subfile, element, and group of elements. There are basically two types of groups, master groups and repeating groups:

Master Group. A master group has no peer or parent group. This logically occurs first in the data structure, and has the further characteristic of being non-repeating; e.g., "NATION" in Group A of Figure 1. Other terms used for this concept are fixed set, master segment, and main-level object.

Repeating Groups. A repeating group is one for which there can be recurring instances of different values for the data elements comprising the group. This type of group may have an arbitrary number of instances and three different relationships with other groups. First, a group may be subordinate in the logical structure to either the master group or another group. (It is important to note, however, that a group may have one and only one parent in a pure hierarchy.) Second, a group may have a peer relationship with one or more groups; that is, several different and logically distinct groups may occur on the same level in the hierarchy. Third, a group may be superior to one or more other groups. Groups B, C, and D, in Figure 1 are examples of repeating groups. (Another term used for this concept is periodic set.)



TYPE	GROUP	DESCRIPTOR	RELATIONSHIP
MASTER	A	NATION	SUPERIOR TO GROUPS B, C, AND D
REPEATING	B	POLITICAL SUBDIVISION	SUBORDINATE TO GROUP A; SUPERIOR TO GROUPS C AND D
REPEATING	C	CITIES	SUBORDINATE TO GROUP B PEER TO GROUP D
REPEATING	D	COUNTIES	SUBORDINATE TO GROUP B PEER TO GROUP C
REPEATING	E	WARDS/ PRECINCTS	SUBORDINATE TO BOTH GROUP C AND D

FIGURE :
LOGICAL STRUCTURE

Logical File

A logical file is the description or specification of the logical aggregation of related groups--one master type and possibly one or more subordinate groups--that will be used to construct a file. Figure 1 is an example of a logical file.

Entry. A particular set of values for all the data elements of a logical file is called an entry. Figure 2 is an example of an entry. Other terms used for this concept are object, logical record, and statement.

File

A file is comprised of one or more entries that have the same logical organization, each of which may differ in physical format (i.e., number of data element values) because of the nature of repeating groups. Entries for all countries, similar to that in Figure 2 for the United States, would constitute a file.

Data Base

A data base is the aggregation of all the files which can be accessed by a data management system.

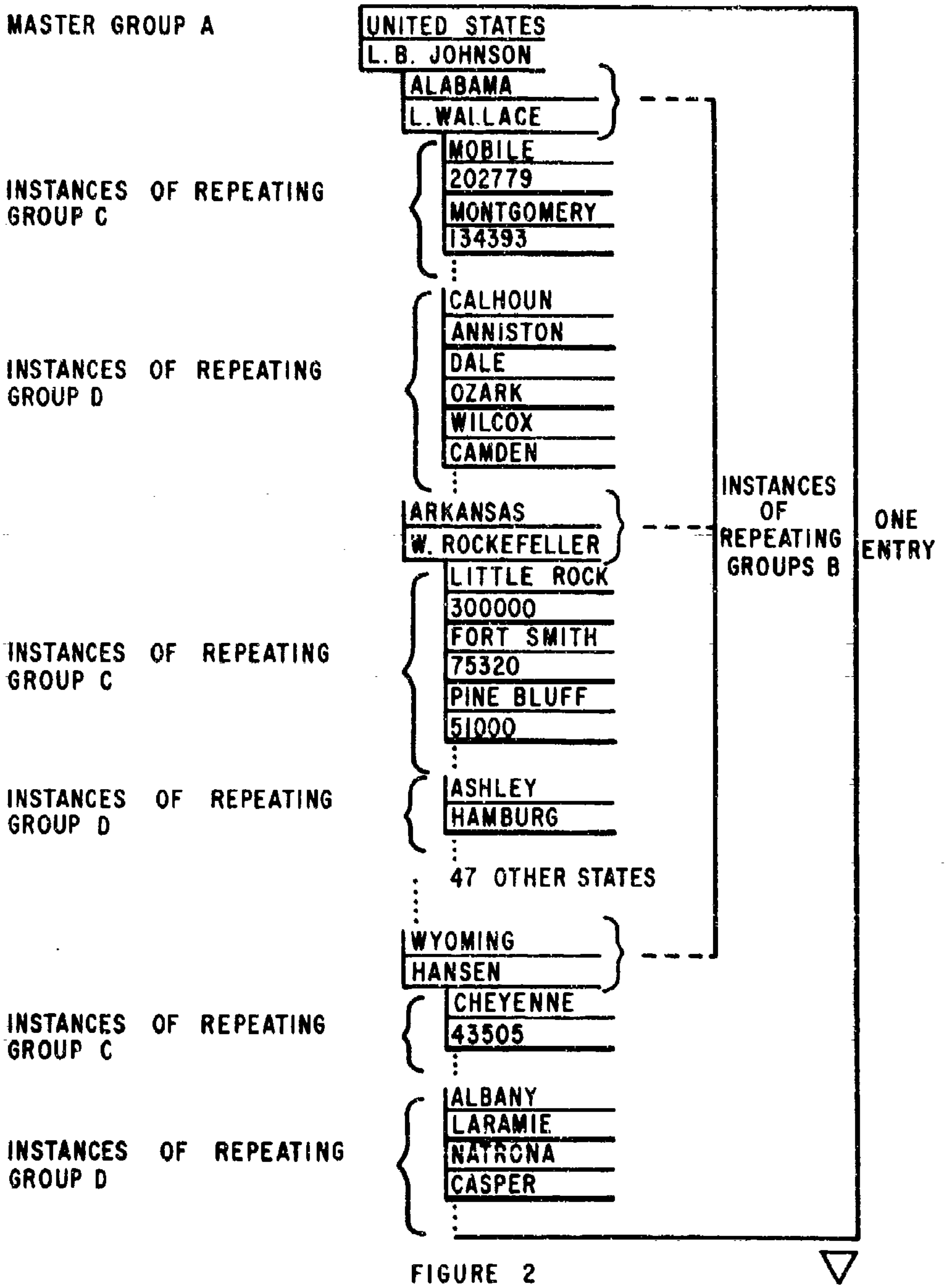


FIGURE 2
AN ENTRY

USE OF THE DATA

It is important to note that the organization of the forms presenting the results of the survey (Section II) does not imply any indication of the relative merits of the various capabilities. The survey is not an attempt to evaluate these systems, but to present the capabilities of the systems in a common framework. It is obvious that using the data for evaluation purposes by any simple approach such as adding up the "yes" and "no" entries would not be valid because some "yes" entries imply "no" in another capability. Also, attempting to establish a generalized weighting system would not be valid because, depending on a specific user's requirements, a "yes" could be a negative factor in a system, and a "no" a positive factor.

As a guideline, it is recommended that the user establish a list of required capabilities using the format of the forms, but completely independent of the answers in this report. Based on the desirability and the required level of implementation of the various features, a comparison with specific systems could yield some useful results, including perhaps a reduction to a shorter list.

SECTION II

SURVEY RESULTS

SURVEY GROUND RULES

To present the responses to the capability list in a common time-frame and with overall consistency, the following ground rules and guidelines were observed throughout the survey.

The time-frame chosen was that the initial capability of the data management systems surveyed must be demonstrable by the second quarter of calendar year 1969. The following terms (see Table I) were used to indicate the vendor's responses: ALL, AV(F), AV(O), BY, CH, K, LIM(O), NA, NF, NO, NTL and YES.

To indicate the final capability of the data management systems, the term AV(F) (available in final version) from the abbreviation list was used. In the particular case that a capability was being considered at the time of the interview but no firm commitment had been made to implement it, the abbreviation NF (not firm) was used. Whenever possible the exact expected dates of initial and final version of the system are given in the forms.

The following guidelines were used for those data management systems which used other processors (e.g., COBOL, ALGOL, PL/I) in part or in whole.

(a) If the data management system or a compiler provides the capability in a single statement, the answer is "yes".

(b) If the capability is not provided by either the system or the compiler, the answer is "no".

(c) If two or more compiler statements, a procedure, are required to provide the capability, the answer is "no" with a footnote reference. This excludes all library subroutines.

TABLE I

LIST OF ABBREVIATIONS

ALL	The entire list is included in the system
AV(F)	Available in the final version
AV(O)	Available as an option through the operating system
BY	Byte--eight-bit representation of characters
CH	Character--six-bit representation of characters
K	Kilo--2^{10}
LIM(O)	Limited--available by operating system and limited by it
NA	Not applicable--because of the particular implementation philosophy, this capability does not have meaning
NF	Not firm--currently being considered although no firm commitment was made at the time of the interview
NO	No--capability is not available
NTL	No theoretical limit--practical device and implementation considerations are noted where known
YES	Yes--the system has the particular capability

(d) If the capability exists through the use of a subroutine library, the answer is "yes" with a footnote reference.

(e) If a special-purpose program, independent of the compiler or data management system, is required to provide the capability, the answer is "no" with a footnote reference.

ORGANIZATION OF SURVEY RESULTS

The results of the survey are presented in a tabular format. The capabilities are listed and an answer column is provided for each system surveyed. It should be noted that the list of the capabilities does not imply any ranking or indication of the relative merits of a particular feature.

An effort to organize the various capabilities into related broad categories caused some problems in answering the questions. For example, the logical structure cf a file is differentiated from its physical structure, and the output presentation section is independent of the retrieval capabilities. In some of the systems surveyed these and other arbitrary divisions were not clearly separated or independent. Whenever possible, the questions were designed to determine the degree of the implementation of a specific capability.

The capabilities and data on availability of data management systems have been enumerated under eight categories:

1.000	File Definition and Organization
2.000	Data Retrieval
3.000	Maintenance
4.000	Output Presentation
5.000	Data Access Methods
6.000	General Interface Capability
7.000	System Statistics
8.000	System Availability

1.000 FILE DEFINITION AND ORGANIZATION

File organization is the foundation on which a data management system (DMS) is built and is the most important aspect of a DMS. It is the major indicator of performance because it dictates the search strategy invoked during retrieval of data. The physical aspects of file organization can have a major effect on the size of the data base; the logical aspects can affect the amount of repetition of the data values and the duplication of data entries.

This section describes the file organization of a DMS with respect to: its logical scheme of organization, the physical structure or layout of the data, the way in which data are defined, and the indexing capability of the DMS.

1.100 Logical File Organization

The logical file organization is the method used to aggregate and organize data. Related data elements, the basic entities in the file (see Section I, Survey Terminology), are collected into groups, and logically related groups are aggregated into logical files. Within a logical file, groups may be ordered according to their relationship to each other, i.e., superior, peer, or subordinate. A number of logical files constitute a data base.

Logical file organization is important in retrieving "information sets" which satisfy retrieval search criteria on the basis of the timeliness and meaningfulness of the information contained therein.

Logical file organization is divided into two broad generic categories: Hierarchical and Heterogeneous.

1.110 Hierarchical Files

A hierarchically structured file is a file that has a specific type of logical group relationship (commonly called tree structure) containing one or more levels. A pure tree-structured hierarchy starts with a single group at the root or base of the structure; this group is defined as the master group, and is unique in the hierarchy. From the master group (or any other group), one or many groups may branch out. However, a given group may have only one immediately superior group. Groups at the same level in the tree are peers of each other (by definition, only one group may exist at the master group level). Groups branch out into many other subordinate groups. A logical group instance is one set of all the data element values in that group.

1.120 Heterogeneous Files

Heterogeneous files are files that do not have a pure tree-structured hierarchy. Logical linkage is allowed between groups of any level, and a group may be subordinate to one or more superior groups. Heterogeneous files are characterized by two additional features: Inverted Tree Structures and a Variety of Logical Structures.

The extent of the logical organization of a DMS is characterized by the following:

1.110 HIERARCHICAL FILE ORGANIZATION	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Data Elements Per Group - The maximum number of data elements that can be included in a group. In the example in Section I the group CITIES may contain only two data elements: CITY NAME and POPULATION.	NTL	(B1)	NTL
Logical Group Instances - The maximum number of instances of a logical group, where an instance is a particular set of data element values of the group.	2^{24}	(B2)	NTL
Levels of Subordination - The depth of the logical structure in terms of the total number of levels of logical groups. The following example represents five levels of subordination: POLITICAL DIVISION CITIES WARDS PRECINCTS STREETS HOUSE NUMBERS	32	1	NTL
Distinct Peer Groups - The maximum number of groups that can occur at the same level of subordination. (See CITIES and COUNTIES in Section I.	254	(B1)	NTL
Related Groups - The maximum allowable number of groups at all levels of subordination that constitute a logical file. In the Section I example, four groups constitute the COUNTRY file: NATION, POLITICAL SUBDIVISION, CITIES, COUNTIES and PRECINTS.	32^{254}	(B1)	NTL

(B1) Size collectively limited to 9999 characters.

(B2) Limited to 999,999.

(E1) Limited by total number of groups.

(F1) The upper bound is limited by the Data Definition Table and/or core.

(F2) Actual limit is 2^{24}-1 which could never be achieved due to record size limits.

(G1) File structure imposes no limit, however, processing techniques chosen by the user may constrain the number of levels.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U. S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NTL	NTL	(F1)	NTL	1023	(I1)	1024
NTL	NTL	NTL(F2)	NTL(G1)	8191(H1)	1	63
NTL	9	15	1	15	0	1
NTL	99(E1)	255	255	255(H2)	0(I2)	1024
NTL	99	(F1)	255	255(H2)	1	1024

(H1) Designed maximum capacity. In initial implementation hardware and software restrictions make this limit unattainable.

(H2) Total number of distinct repeating groups per file is limited to 255.

(I1) Record length 1023 characters. Theoretically each character could be defined as a data element.

(I2) No level of subordination; therefore no peer relationship.

*U. S. ARMY SYSTEM DEVELOPED BY CDC

1.110 HIERARCHICAL FILE ORGANIZATION (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Entries - The maximum number of physical entries in a file. A physical entry is the collection of all instances of the subordinate groups subsumed under one instance of a master group. In the Section I example, the COUNTRY file would contain an entry for each country in the data base.	2^{24}	(B1)	NTL
1.120 HETEROGENEOUS FILE ORGANIZATION			
Inverted Tree Structure - The capability to have an "inverted tree structure" and have a subordinate group linked to more than one superior group at any level, as illustrated by the example in Figure 1 of Section I.	YES	NO	NF
Variety of Structures - The capability to define a logical file structure with null valued data elements (data elements containing either no data values or only parts of group instances at generation time). The implication here is that there is no "penalty" for defining future data elements in the file; i.e., no physical storage is allocated in advance. There are two cases:			
- Missing Groups - Groups which have been defined have no instances, e.g., no values for CITY NAME or POPULATION	YES	YES	NF
- Missing Data Elements - Data elements which have been defined have no value, e.g., CITY NAME filled in but no values for POPULATION	YES	YES	NF

(B1) Limited to 999,999.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NTL	NTL	NTL	LIM(O)	16,777,215 (H1)	NTL	2^{23}
YES	NO	NO	NO	NO(H2)	NO(I1)	NO
YES	YES	YES	YES	YES	YES	YES
NO	NO	NO	YES(G1)	YES	YES	YES

(G1) Only variable fields.

(H1) Designed maximum capacity for all instances of all groups (entries if there are no repeating groups) within a file. In initial implementation hardware & software restrictions make this limit unattainable.

(H2) This can be done by restricting the file and inverting the groups.

(I1) No level of subordination, therefore no "inverted" relationship exists.

*U.S ARMY SYSTEM DEVELOPED BY CDC

1.200 Physical File Organization

A physical file is a file of data valucs (not necessarily reflecting the logical organization of the data base). There are various methods for physical organization of data. In general the methods are indicators of the response time of the system, i.e., the time it takes to retrieve data from the data base. Two basic types of physical file organization are discussed: Parallel and Sequential.

1.210 Parallel

This method collects and stores all the values for a particular data element or group instance in a contiguous manner. It is independent of the logical organization. Physically contiguous is interpreted to be independent of device considerations such as track size. Usually physical links are used to preserve the hierarchical organization.

1.220 Physical Sequential

The natural way of storing hierarchically structured data is with each group expanded according to its position in the hierarchy.

The physical file organization of the DMS is depicted by the following categories.

1.210 PARALLEL PHYSICAL FILE ORGANIZATION	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
By Data Elements - All values of a data element are stored contiguously. For example: All the values of POPULATION for all cities in all states are stored contiguously independent of the group instance or entry they are part of.	NO	NO	NF
By Logical Groups - All instances of a logical group are stored contiguously. For example: All the values for CITY NAME, and POPULATION for all cities in all states are stored contiguously. Note: The example of the Nation file in Section I is not a parallel structured file.	NO	NO	NF
1.220 SEQUENTIAL FILE ORGANIZATION			
By Group Instance - All values of the data elements comprising a group instance are stored contiguously. For example, all the values for CITY NAME and POPULATION for the state of Alabama are stored contiguously as shown in Section I.	YES	YES	YES(C1)
By Hierarchical Group - Each group instance is stored according to its subordinate relationship, starting with the master group. Each and every group is stored according to the group instance discipline. First, the instance of the master group is stored, then an instance of each successively subordinate group in the hierarchy is stored accordingly until the terminal group is reached. Next, all instances of the terminal group are expanded and stored. If there is no peer at the terminal group level, the next highest group instance is expanded. If there are peer groups at the terminal group level,	YES	YES	YES(C1)

(C1) When a file is divided into multiple data sets, different levels of hierarchical records may be physically stored on different devices.

(D1) By data declaration.

(F1) If desired, individual groups can be defined as a separate file and accessed in a single retrieval by the multiple file capability.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO	NO	NO	YES	NO	NO
YES(D1)	NO	NO(F1)	NO(G1)	NO	YES(I1)	YES
YES(D1)	YES	YES(F1)	YES	NO	YES	YES
YES(D1)	YES	YES	YES	YES(H1)	YES(I1)	YES

(G1) When a file consists of only master groups.

(H1) Actual values are stored contiguously when they are dates or numbers requiring less than 22 bits; otherwise the actual value is stored separately (one time only) and a pointer is used.

(I1) Special-case logical file consists of only master groups.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

1.220 SEQUENTIAL FILE ORGANIZATION (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
By Hierarchical Group (continued) these are expanded in their entirety before moving back up the tree. The process continues until the master group is completely expanded, ending the process. For example in Section I, the COUNTRY NAME and its PRESIDENT are stored first, followed by the values of STATE NAME and GOVERNOR for first state, followed by all the values of CITY NAME and POPULATION for the first state, followed by all values of COUNTY NAME and COUNTY SEAT for the first state, followed by the values of STATE NAME and GOVERNOR for the second state, etc.			
By Entry - Every entry is stored contiguously. For example, all the values in the entry UNITED STATES are stored contiguously, followed by all the values for the entry URUGUAY. The example shown of the NATION file in Section I is a physical sequential file.	YES	NO	YES(C1)

(C1) When a file is divided into multiple data sets, different levels of hierarchical records may be physically stored on different devices.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES(G1)	YES(H1)	YES	YES

(D1) By data declaration.

(G1) ISAM (Index Sequential Access Method).

(H1) Actual values are stored contiguously when they are dates or numbers requiring less than 22 bits; otherwise the actual value is stored separately (one time only) and a pointer is used.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

1.300 Data Definition

Data definition is the specification of the types of data representation acceptable to the DMS and the physical limitations on quantity and content.

1.310 Data Element Representation

Various types of data element provided by the DMS to the software user, such as fixed-point and floating-point numbers, are described in machine-oriented terms.

1.320 Physical Size

The lengths of the various types of data element provided by the DMS to the software user are also described in hardware terms. Thus, many computers are capable of processing floating-point numbers 32 bits in length, as well as floating-point numbers 64 bits in length; the DMS's for such computers will therefore offer the user the ability to define either long (64 bits) or short (32 bits) floating-point data elements.

1.330 Storage Devices

Depending on computer haidware configurations and DMS capabilities, the user will have many different secondary storage devices which he can use for permanent or semi-permanent data-base storage.

1.310 DATA ELEMENT REPRESENTATION	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Fixed Point	YES	YES	YES
Floating Point	YES	YES	YES
Packed Decimal	YES	NO	YES
EBCDIC/BCD	YES	YES(B1)	YES
ASCII	NO(A1)	YES	NF
1.320 PHYSICAL SIZE			
Data Element - The maximum number of characters in data elements.			(C1)
- Variable Length	254 BY	NONE	
- Fixed Length			
● Numeric	254 BY	18dd	
● Alphanumeric	254 BY	99CH	
Group - The maximum number of characters in logical groups.	254^2BY		(C1)
- Fixed Length		YES	
- Variable Length		NONE	
Entry - The maximum number of characters in entries.	NTL		(C1)
- Fixed Length		9999	
- Variable Length		NONE	
File			
- The maximum number of entries per file.	NTL	999999	(C1)
- The maximum number of files per data base.	NTL	(B2)	(C1)

(A1) Any representation can be added to the system.

(B1) Through BCL (Burroughs Common Language).

(B2) Expandable to 99.

(C1) Physical size of all elements depends on COBOL implementation

(D1) Through IDS and COBOL.

(D2) 4096 characters including control characters.

(E1) ASCII character representation is handled automatically by IBM hardware.

(E2) Variable data element lengths are not supported. An entry may contain a variable number of fixed-length groups. A group may consist of one data element.

(E3) 4 bytes (binary or floating point); 16 bytes packed decimal.

(E4) Size of records (entries) is limited by memory space available for buffer. The absolute maximum for either fixed- or variable-length records is 99,999 bytes. This may be increased (overridden) to the maximum memory size in OS/360 by use of the JCL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	NO(F1)	YES	YES	NO	YES
NO	YES	NO(F1)	NO(G1)	YES	NO	NO
NO	YES	YES	NO	NO	NO	NO
YES(D1)	EBCDIC	YES	YES	YES	BCD	YES
NO	(E1)	---	NO	LIM(O)	NO	NO(J1)
NONE	(E2)	NONE	NTL(G2)	255BY	NONE	5629
(D2)						
	(E3)	31dd	4BY	255BY	1023	5629
	255	255BY	255BY	255BY	1023	5629
(D2)		LIM(O)(F2)				
	9999		(G3)	295805(H1)	1023	5629
	NONE		NTL	295805(H1)	NONE	5629
NTL	(E4)	LIM(O)(F3)	(G4)			
				295805(H1)	1023	5629
				295805(H1)	NONE	5629
NTL	NTL	NTL	LIM(O)	16,777,215 (H1)	NTL	NF
NTL	NTL	NTL	NTL	1	NTL	NTL

(F1) Can be defined and passed to processing subroutine for action and/ or conversion.

(F2) Group is fixed length. The length is limited to the track size on the direct-access device (unless track overflow feature is available) or the maximum record size from tape.

(F3) Entry is variable length only. Using physical sequential access, length is track- or record-limited.

(G1) May be carried, but not processable.

(G2) Block size (physical block size) is generation parameter.

(G3) Block size core 1K.

(G4) Restricted by size of buffer within the different FFS modules min 8K to max size of buffer allocated.

(H1) Designed maximum capacity. In initial implementation hardware and software restrictions make this limit unattainable.

(J1) Available as a hardware option.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

1.330 STORAGE DEVICES	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Tape	(A1)	NO	YES
Disc			
- Pack	(A1)	NO	YES
- File	(A1)	YES	YES
Drum	(A1)	NO	YES
Data Cell	(A1)	NO	YES

(A1) DM-1 is indifferent to the types of storage devices.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	(E1) 2400	2400(all)	YES	NO(H1)	YES	NO
YES	2311	2311	YES	YES	NO	YES
YES	2302	2314	YES	YES	NO	YES
YES	2312	YES	LIM(O)	NO	NO	NO
NF	2321	2321	LIM(O)	NO	NO	YES

(E1) System uses DOS or OS data management facilities and will use any storage media supported by IBM.

(H1) Used for backup storage and warehousing, but not for active access.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

BLANK PAGE

1.400 Indexing Schemes

Indexing, a characteristic of data files, provides the means to retrieve data from the file by associating a specific physical location with every logical data aggregate of the file--that is, indexing allows a system user or a computer program to determine where any particular logical part of the file is physically located. Whether a user or a computer program can efficiently determine physical location is another matter: efficiency depends on the indexing scheme used as well as on the structure and contents of the file, the characteristics of the secondary storage device on which the file resides, and the quality of the DMS. Four indexing schemes are discussed here.

1.410 Name Index

This indexing method associates physical locations with unique symbolic names. It is useful when a file component whose location will be requested is always uniquely identified by a symbolic name (i.e., does not require further qualification by data value or ordinal location within a repeating set). For example, if a file contains a data element "COUNTIES" which is an element of a repeating group, and the file is name-indexed, then the indexing scheme can provide the (starting) location for the storage assigned to "COUNTIES." The name-index scheme could not, however, provide a physical location for "COUNTIES(17)" or for "COUNTIES WHEN CITIES = 'ANNAPOLIS'" (except by non-responsively saying something like "COUNTIES STARTS ON TRACK 4 OF CYLINDER 73 OF DISC 22222"). On the other hand, a file containing library subroutines would be indexed quite well by name indexing.

1.420 Device-Element Index (Indexed Sequential)

This indexing method associates physical locations with every ordinally distinct file-component instance, although file-component values are not usable for indexing purposes. Thus, the indexing scheme could provide the physical location of the data value of the second instance of "CITIES" corresponding to the seventh instance of "COUNTIES" corresponding to the 42nd instance of "STATES" for the third instance of "NATIONS." If the city in question were "ANNAPOLIS," the device-element indexing scheme could provide the information that CITIES(2, 7, 42, 3) was at track 17 of cylinder 191 of disc 22222; but this scheme would not be able to supply the location of "CITIES WHERE CITIES - 'ANNAPOLIS' ". Thus this method provides more detailed indexing than the name-indexing scheme described above, but at the cost of much larger index tables and the longer retrieval times involved in processing them.

1.430 Data-Block Index (Sequential)

This indexing method associates ordinally distinct file-component instances with physical data blocks of fixed size, which can be thought of as the pages of storage device volume. The index is implemented physically as a table of contents, omitting the structural indentations which are found in tables of contents in books and magazines, with an individual "chapter" listing for each page of the physical file. The method is used mainly for data files which are stored on sequential-access secondary-storage devices, such as magnetic tape and paper tape, although it can also be used for a data file on an immediate-access device such as a disc or drum. In order to find the location of a desired logical file component, one must sequentially search the data-block index for the name or qualified name of the component; one might think that the index should therefore be inverted into a device element index. In fact, however, obtaining the physical location of a file component is mainly useful as a preliminary to the physical retrieval of the associated file-component value or values, which on a sequential device must necessarily involve retrieving and examining every physical element of the file successively until the desired file component shows up.

1.440 Algorithmic Mapping

This indexing method associates physical locations with data values. A single-value function which takes the data value as its argument is used to produce a unique physical address (an additional operation must be performed to see that duplicate values are placed in different storage locations, but this is usually a minor correction). The data value which is used to obtain the storage location may be the entire file component, or it may be only a part of the component (e.g., when storing an entire group instance or an entire entry, a single data element value in the group or entry may be used to produce the physical address).

1.410 NAME INDEX	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
The symbolic name which is used as an index may be that of any of the following types of data aggregates:			
- The name of the file itself	YES	YES	YES
- The name of a group of the file	AV(F)(A1)	NO	NF
- The name of a data element of the group	AV(F)(A1)	NO	NF
1.420 DEVICE ELEMENT INDEX (INDEXED SEQUENTIAL)			
The types of secondary storage devices which can be used in conjunction with this indexing method are:	(A2)	NO	ALL
- Disc - Data Cell - Drum			
1.430 DATA BLOCK INDEX			
Data block size (in bits, bytes, or characters)	AV(F)(A3)	NA	NF
1.440 ALGORITHMIC MAPPING			
The types of file components which can be stored as entities when indexing via algorithmic mapping are:			
- File	AV(F)(A4)	NO	NO
- Entry	AV(F)(A4)	NO	NO
- Group	AV(F)(A4)	NO	NO
- Group Instance	AV(F)(A4)	NO	NO
- Data Element	AV(F)(A4)	NO	NO

(A1) By April, 1969.
(A2) DM-1 index address is logical address, not physical address; therefore it is device-independent.
(A3) By April, 1969 DM-1 will have hierarchical block location tables.
(A4) Hash coding expected in 1969.
(D1) IDS is a device-independent and deals with record identifiers which are available to the user and which the system maps into physical addresses without any mass storage access.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NA(D1)	YES	YES	YES	YES	NO	YES
NA(D1)	NO	NO	NO	NA	NO	YES
NA	NO	NO	NO	YES	NO	YES
NA	(E1)	(F1)	(G1)		NO	
				AV(O)		YES
				NA		NO
				NA		NO
NA	(E2)	NA	NA	NA	NA	NA
YES	NO	NA	NO	NO	NO	NF
NO	NO	NA	NO	NO	NO	NF
YES	NO	NA	NO	NO	NO	NF
YES	NO	NA	NO	NO	NO	NF
YES	NO	NA	NO	NO	NO	NF

(E1) ISAM (Index Sequential Access Method).

(E2) MARK IV retrieval is based on a one-pass-of-the-master file concept. Each record is read into a buffer and then the address of each data element in the record is available both to own code and the system.

(F1) ISAM (Index Sequential Access Method).

(G1) ISAM (Index Sequential Access Method).

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.000 DATA RETRIEVAL

Retrieval is the process of extracting data element values from those entries which satisfy user-specified search criteria. Retrieval is not an end in itself, but is useful only as a service capability for the other DMS facilities such as maintenance and output presentation. A DMS user will not specify directly that retrieval should occur; rather, he will request that a certain maintenance or output operation take place, and that the data required by the operation be retrieved according to the search criteria which he specifies.

Consider these simple examples:

```
Maintenance: CHANGE CLASS TO "CITY" IF
             POP GT (25000)
```

The statement as a whole is a maintenance request statement, while the phrase "IF POP GT (25000)" is a retrieval "request" sub-statement or clause.

```
Output Presentation: DISPLAY GROUP "CITIES" IF
                     POP GT (25000)
```

In this case we have the same retrieval "request" clause or sub-statement as above, although the entire statement is an output request.

Note that in neither case is the user primarily interested in forming a subset (of the data file being used) which satisfies the criteria expressed in the retrieval clause. Rather, his primary interest is in modifying the data base in the one case, and in observing the contents of the data base in the other case. For these examples a very simple type of search criteria is used.

In the general case, a retrieval criteria specification consists of one or more retrieval statements. The material in the following paragraphs covers the construction of retrieval statements at three levels:

(a) the structure of the retrieval statement; i.e., the types of expressions which can be used as components

(b) the sub-structure of the retrieval statements; i.e., the combination of operators and operands into compound expressions

(c) the super-structure of the retrieval statements; i.e., how retrieval statements are organized into a retrieval criteria specification.

2.100 Retrieval Expressions

Retrieval expressions are the primary structural elements of retrieval statements. There are two types of retrieval statements useful in constructing a retrieval specification: "operational" retrieval statements and "control" retrieval statements.

An operational retrieval statement consists of one or more expressions; an expression consists of an operator and one or more operands. The general form of an expression is then:

(operand/operator/operand) for binary operators, and

(operator/operand) for unary operators.

Construction of operational retrieval statements is considered with respect to the aggregation of their basic components from the simplest to the more complex statements. Operands combine with operators to form simple expressions. Simple expressions are punctuated with parentheses and are combined by logical connectors to form compound expressions. Compound expressions are aggregated to form retrieval expressions, and retrieval expressions are collected to form operational retrieval statements. Executional control of operational retrieval statements is provided by control retrieval statements such as GO TO, DO, etc.

2.110 Conditional Expressions

Conditional, or logical, expressions form the building blocks of operational retrieval statements. A logical expression is a special case of the simple expression in which the operator is a logical operator. Conditional expressions state a relationship between operands.

2.111 Operands

An operand is a data entity upon which an operation is performed; it may be a simple operand or a compound operand. A simple operand may be a data element, a literal value, or the results of some computation. A compound operand is a combination of simple operands.

EXAMPLES: IF "AGE" GT (25)--(25) is a simple operand;

IF "AGE" BETWEEN (18, 25)--(18, 25) is a compound operand.

When the values of two data elements are compared, the two operands may be different data elements or the same data elements. They may be selected from the same logical entry, or from two different logical entries.

2.112 Operators

Logical operators are the linguistic devices a DMS provides for expressing the relationship between operands. The general form of a logical expression will be:

(operand/operator/operand) for binary operators, and

(operator/operand) for unary operators.

In addition, simple logical expressions can be connected to form compound logical expressions (see Section 2. 130).

2. 120 Arithmetic Expressions

An arithmetic expression is another general type of retrieval expression whose operators may be arithmetic operations or mathematical functions. This category describes the specific computational capabilities of retrieval expressions and considers specifically:

(a) types of operands

(b) basic arithmetic operators

(c) mathematical functions

(d) modes of arithmetic expressions.

2. 130 Complexity of Expressions

The level of complexity of expressions indicates the number and variety of combinations of expressions into operational retrieval statements. In this category the following topics are considered:

(a) compound logical expressions

(b) levels of nesting within expressions

(c) quantitative limitations on formation of compound expressions from simple expressions

(d) mixed mode expressions.

2.110 CONDITIONAL EXPRESSIONS	AUERBACH	BURROUGHS	CSC
2.111 OPERANDS	DM-1	FORGE	COGENT III
The following items indicate the capability for comparing the values of data elements selected from different structural subdivisions of a file, specifically:			
Data Element - Comparison of the values of two different data elements in the same group of the same entry. EXAMPLE: "Country of Birth" EQ "Country of Citizenship"	YES	YES	YES
Literal EXAMPLE: "Country of Birth" EQ (USA)	YES	YES	YES
Results of Computation - The computation of a single value from some mathematical combination of selected data element values, and the comparison of the computed value against the data element. EXAMPLE: "DIST" GT SQRT ("AREA"/(3.14)) or X = SQRT ("AREA"/(3.14)) "DIST" GT X	AV(F)(A1)	YES	YES
Different Group Instance, Same Data Element EXAMPLE: "POP-COUNTY(2)" GT "POP-COUNTY(3)"	NO	NO(B1)	YES
Different Entry, Same Data Element - The ability to compare the same data element in two different file entries.	NO	NO(B1)	YES

(A1) Computational functions to be added in 1969.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES(H1)	YES	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	YES	YES	NO	YES(H1)	NO	YES
YES(D1)	YES	YES(F1)	NO	NO	NO	NO
YES(D1)	YES	YES(F1)	NO	NO	NO	NO

(D1) Available through IDS and COBOL.

(F1) Using auxiliary user-defined work area.

(H1) Included in language design, but implemented in only part of the system initially.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.112 OPERATORS	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Standard Comparators - The "standard" operators are those which are generally available in every DMS: - EQ Equal - NE Not equal - GT Greater than - LT Less than - LTE Less than or equal to - GTE Greater than or equal to	ALL	ALL	ALL
Additional Comparators - Additional operators are those which are used in many systems in order to reduce the size and complexity of the retrieval statements needed to perform complex retrievals:			
- Between - The value of one operand is within a range of the values specified in the second operand. EXAMPLES: "AGE" between (19, 25) can be equivalent to "AGE" GT (18) AND "AGE" LT (25), or "AGE" GTE (18) AND "AGE" LT (25), or "AGE" GT (18) AND "AGE" LTE (25), or "AGE" GTE (18) AND "AGE" LTE (25) depending on whether the operator is defined as inclusive or exclusive of the particular values.	YES	NO(B1)	YES

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
ALL(D1)	ALL	ALL	ALL	ALL	ALL	ALL
YES(D2)	NO	YES	YES	YES	NO	YES

(D1) Available through COBOL.
(D2) Available through IDS and COBOL.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.112 OPERATORS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
- GT Not Blank - Greater than excluding blanks; in order for the comparison to succeed, the value of the first operand must be not blank and it must also be greater than the second operand. EXAMPLES: PROFIT GT-NOT-BLANF (-1000000) is equivalent to PROFIT NE (Ø) AND PROFIT GT (-1000000) (For this operand, the term "blank" means a literal blank or a numerical zero).	YES	NO(B1)	NF
- LT Not Blank - Less than excluding blanks; analogous to the previous operator.	YES	NO(B1)	NF
- Character Pattern - The characters or bytes of the first operand occur as a substring within the second operand. EXAMPLES: (ABC) character-pattern (ZXABCWT) gives a successful test result. (ABC) character-pattern (ZXABWCT) gives an unsuccessful test result.	NO	NO	NF

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NO	(F1)	YES	NO	NO	NO
YES(D1)	NO	(F1)	YES	NO	NO	NO
NO	NO	YES	YES	NO	NO	NO

(D1) Available through IDS and COBOL.

(F1) Available through compound conditions.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.112 OPERATORS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- Mask Match - The mask match is the name of a class of operators which have certain common features: certain positions of an operand are examined or are not examined, depending on whether the corresponding position in a mask contains a "search" indicator rather than a "don't search" indicator such as an "*"; each position of the operand which is examined is tested, either for the presence of a specified value, or against the corresponding position of a second operand or against the character specified in the mask. The test may be deemed successful if all of the positions which are examined have individual test successes (AND match), or if at least one of them has an individual success (OR match). EXAMPLES. (*A*D**) AND-MASK-MATCH (ABCDEF) succeeds. (*B*D**) AND-MASK-MATCH (ZBXDVF) succeeds.	NO	NO(B1)	NF

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NF	YES	YES	YES	NO	NO

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.112 OPERATORS (Continued)	AUERBACH	BURROUGHS	CSC
	DM 1	FORGE	COGENT III

- <u>Maximum/Minimum Value</u> - If a group contains one or more instances of a variable; then the test succeeds if the value of the operand is the largest (or smallest) value for that variable of all those in the group.
EXAMPLES: If there are five instances of AGE, with values AGE(1) = 28, AGE(2) = 32, AGE(3) = 36, AGE(4) = 36, and AGE(5) = 28, then
max "AGE(2)" fails;
max "AGE(3)" succeeds;
min "AGE(1)" succeeds.
If there is only one instance of AGE, AGE = 18, then min "AGE" succeeds and max "AGE" succeeds.
	NO	NO	NF

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO	(F1)	YES	NO	NO	NO

(F1) Defined as comparing the value of maximum or minimum occurrence to another value: e.g., when "IF MAXIMUM AGE GR 35" succeeds, then "IF MAXIMUM AGE GR 40" fails.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2. 112 OPERATORS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III

- Empty - This test succeeds if the variable specified by the operand has no instance in the file, or if the operand value is zero (for a numeric variable) or blank (for a character variable).
EXAMPLES: If there are no instances of AGE in the file (but AGE is defined as an element of the file), then
empty "AGE" succeeds.
If there are three instances of PROFIT, with values PROFIT(1) = -1000000, PROFIT(2) = 0, PROFIT(3) = 300000, then
empty "PROFIT(1)" fails;
empty "PROFIT(2)" succeeds;
empty "PROFIT(3)" fails.
	YES	NO(B1)	NF

- Increase/Decrease - This test succeeds if a change in value occurs between successive tests of this operator. EXAMPLES: If AGE(1) = 20, AGE(2) = 20, AGE(3) = 17, AGE(4) = 17, AGE(5) = 17, AGE(6) = 28, AGE(7) = 30, then increase AGE(i); testing AGE in sequence beginning with AGE(0) succeeds for i = 6, 7, and fails for i = 1, 2, 3, 4, 5; and decrease AGE(i) succeeds for i = 3, and fails for i = 1, 2, 4, 5, 6, 7.	NO	NO(B1)	NF

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SCS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO(E1)	YES(F1)	NO	YES(H1)	NO	NO
NO	NO	YES	NO	NO	NO	YES

(E1) These can all be ascertained with processing-request statements.

(F1) Two tests are available, one for absence (no data), one for empty (zero or blank).

(H1) The operator is FAILS. Its opposite (EXISTS) is also included in the language.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.112 OPERATORS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- Geographic Search - The capability to locate data element values in geometric areas based on their geographical coordinates.	NO	NO	NF
EXAMPLES: ALL CITIES WITHIN 20 MILES OF NEW YORK ALL SHIPS WITHIN (N36°50', W75°55'), (N37°5', W76°20'), (N36°45', W76°30').			
- Other - This entry in the survey is for specifying any DMS relational operators which do not fit under any of the other categories.	NO	NA	NF
2.120 ARITHMETIC EXPRESSIONS			
2.121 OPERANDS			
Data Elements	(A1)	YES	YES
Constants	(A1)	YES	YES
Results of Computation	(A1)	YES	YES
2.122 OPERATORS			
= Equality (replacement)	(A1)	YES	YES
+ - Addition and subtraction	(A1)	YES	YES
* / Multiplication and division	(A1)	YES	YES
EXP Exponentiation	(A1)	YES	YES
EXAMPLES: "AGE" = "CURRENT DATE" - "BIRTH DATE" "NEW SALARY" = (2) * "OLD SALARY" "GRADE" = "AGE" - (5).			

(A1) Arithmetic operators to be added in 1969.
(D1) Available through IDS and COBOL.
(D2) Available through COBOL.
(E1) Provision is made for user insertion of any desired operations.
(F1) Available through subroutine linkage.
(F2) CHANGE--either increase or decrease; BREAK--similar to change, gives control in reporting on last occurrence of test value.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NO	NO(F1)	YES(G1)	NO	NO	NO
YES(D1)	NO	(F2)	(G1)	NO	NO	NA
YES(D2)	YES	YES	YES	YES	NO	YES
YES(D2)	YES	YES	YES	YES	NO	YES
YES(D2)	YES	YES	NO(G2)	YES	NO	YES
YES(D2)	YES	YES	(G2)	YES	NO	(J1)
YES(D2)	YES	YES	(G2)	YES	NO	(J1)
YES(D2)	YES	YES	(G2)	YES	NO	(J1)
YES(D2)	(E1)	NO(F1)	(G2)	YES	NO	(J1)

(G1) Polygon and circle search capabilities.
(G2) Capability in Output Module.

(J1) Available as a subroutine call to FORTRAN library.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.123 MATHEMATICAL FUNCTIONS	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
The following mathematical functions can sometimes be used as operators:			
Square root	(A1)	YES	(C1)
Natural logarithm	(A1)	YES	(C1)
Trigonometric functions	(A1)	(B1)	(C1)
- sine			
- cosine			
- tangent			
- arcsine, arccosine, arctangent			
2.124 MODE OF COMPUTATION			
This indicates which of the computer hardware arithmetic modes are used in obtaining computed operands:			
Floating point	(A1)	YES	YES
Decimal (character)	(A1)	NO	YES
Integer (fixed point)	(A1)	YES	YES

(A1) Arithmetic operators to be added in 1969.

(B1) Only ARCTAN is available through COBOL.

(C1) Function of COBOL implementation.

(D1) Available through COBOL.

(E1) May be inserted by user.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U. S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	(E1)	NO(F1)	YES(G1)	YES(H1)	NO	NO(J1)
NO	(E1)	NO(F1)	YES(G1)	YES(H1)	NO	NO(J1)
NO	(E1)	NO(F1)	YES(G1)	YES(H1)	NO	NO(J1)
NO	YES	NO	NO	YES	NO	NO
YES(D1)	YES	YES	NO	NO	NO	NO
YES(D1)	YES	NO	YES	YES	NO	YES

(F1) Available through subroutine linkage.

(G1) Available as a system subroutine.

(H1) Also absolute value and integer operators.

(J1) Available as a subroutine call to FORTRAN library.

*U. S. ARMY SYSTEM DEVELOPED BY CDC

	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
2.130 COMPLEX EXPRESSIONS			
2.131 LOGICAL (BOOLEAN) CONNECTORS			
Logical connectors combine logical expressions into operational retrieval statements. There are two binary logical connectors and one unary logical connector.			
The binary logical connectors are:			
AND - Both logical expressions must be satisfied	YES	YES	YES
OR - Either one of the logical expressions must be satisfied (both may be satisfied).	YES	YES	YES
The unary logical connector is:			
NOT - The logical expression must not be satisfied.	YES	YES	YES
2.132 LEVELS OF NESTING			
The number of levels of nesting possible through the use of parentheses to nest two or more logical expressions in order to change the implied logic (i.e., natural precedence of operators) of the retrieval statement.	NTL	NTL(B1)	(C1)

EXAMPLES:
(Zero level nesting = no nesting):
IF "AGE" LT(4) AND "TITLE" EQ (STAFF) OR "AGE" GTE(25) OR "SALARY" LT(9000)
(One level of nesting):
IF "AGE" LT(40) AND ("TITLE" EQ (STAFF) OR "AGE" GTE(25)) OR "SALARY" LT(9000)
(Two levels of nesting):
IF "AGE" LT(40) AND (("TITLE" EQ (STAFF) OR "AGE" GTE(25)) OR "SALARY" LT(9000)).			

(B1) Object Code cannot exceed 1024 core storage words.

(C1) Function of COBOL implementation.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	NO	YES	NO(G1)	YES	YES	YES
NTL	9	4	8	NTL	0	1

(D1) Available through COBOL.

(G1) The NOT is available at the logical operations level.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.133 COMBINED EXPRESSIONS	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
The number of simple expressions that can be combined in an operational retrieval statement:			
Logical - Whether there is a limit to the number of logical expressions which can be combined into a single operational retrieval statement, and if so, what the limit is. The combinations considered here are those which are found using the logical connectors AND, OR, NOT.	NTL	NTL	NTL
Arithmetic - Whether there is a limit to the number of terms which can be combined into a single computed operand, and if so, what the limit is. The combinations considered here are those which involve arithmetic operators (2.122) only, not mathematical functions (2.123).	AV(F)(A1)	NTL(B1)	NF
EXAMPLE: Z = (A + B/C = Q + 10 - D/2)/10 has eight operations.			
2.134 MIXED ARITHMETIC/BOOLEAN EXPRESSIONS			
Whether an operational retrieval statement may combine both the arithmetic and Boolean operators, compute the intermediate results according to the correct operator precedence relationship, and successfully combine these results to evaluate the total expression.	NO	YES	YES
EXAMPLE: 2* "DIST" GT SQRT (3* "AREA"/ (3.1416)) AND "POP" + 10000 GT (100000).			

(A1) When arithmetic operators are included.

(B1) Object code cannot exceed 1024 core storage words.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NTL	NTL	253	NTL	NTL	NTL	100
NTL	1(E1)	253	NTL	NTL	0	NTL
YES(D1)	YES	YES	NO(G1)	YES	NO	YES

(D1) Available through COBOL.

(E1) Results of one operation may be used in the next operation (i.e., a + b = c; c + x = y).

(G1) Capability in Output Module.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.200 General Language Capabilities

The construction of individual operational retrieval statements has already been considered in some detail. The DMS capabilities referred to in this category are those which permit the user to cause actual retrieval to occur, and to control the resulting retrieval operations. They consider:

(a) what retrieval operations can do (general)

(b) what control statements can do within retrieval specifications

(c) capabilities for using extra-DMS facilities within a retrieval specification

(d) capabilities of retrieval specification which are important during output presentation of data from files.

2.210 General Retrieval

The general retrieval capabilities of a DMS are those which are implicitly rather than explicitly specified by the syntax and semantics of the DMS retrieval language, and which therefore can provide the user of a DMS with powerful retrieval capabilities whose existence might not be immediately evident. This category considers the following topics:

(a) the sources of data elements used as operands in logical expressions

(b) the destinations of such data elements

(c) the sources of user requests for retrieval of such data elements.

2.220 Procedural Language

This indicates the capabilities which are available, using control retrieval statements, to affect the flow of processing through the retrieval statements of a retrieval specification.

The individual statements of a retrieval specification do not necessarily have to be executed sequentially; rather, the execution sequence can be made to depend on requirements for efficiency in the construction of retrieval specifications, or on the nature of the data being retrieved. Thus, considerable flexibility and sophistication can be achieved in the retrieval process.

2.230 Language Interface

This indicates the capability to use a control retrieval statement to specify processing of a retrieval specification in another form or "language." In this case the control statement is embedded utilizing facilities beyond those normally considered to be part of a DMS. Two specific capabilities are considered:

(a) the use of retrieval statements written in a language other than that of the DMS

(b) the execution of library subroutines

2.240 Other Language Capabilities

This deals with the specification of a number of control retrieval statements which allow the DMS user to produce or create data values and retrieval specifications at one point in a sequence of data management operations, and then to make further use of them at a later time (e.g., when output presentations are made). This category considers the following topics:

(a) temporary storage of retrieved data

(b) creation of prestored retrieval specifications

(c) modification of prestored retrieval specifications

(d) parameterized execution of prestored retrieval specifications

2.210 GENERAL RETRIEVAL CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Retrieve Elements - Whether a system has the capability to extract (retrieve) any data element value from any group instance, master or repeating, in the file independently of the particular level at which it resides.	YES	NO(B1)	YES
Search Elements - Whether a system provides the capability of extracting values for those data elements used as operands in the logical or arithmetic expressions within retrieval statements.	YES	NO(B1)	YES
Non-Search Elements - Whether a system provides the capability of extracting values for data elements other than those used as operands in logical expressions; in other words, the ability to extract a group instance or entry based on a particular data element value.	YES	NO(B1)	YES
Simultaneous Retrievals - Whether the system provides the capability of retrieving data from two or more files with the same logical format or files with some common data elements within the same computer run.	AV(F)	NO(B1)	YES
Specific Group Instance - Whether the user may specify an ordered instance of a repeating group he wished to extract. EXAMPLES: Retrieve specified instance of the group DEPENDENT(1) Retrieve specified instance of the group DEPENDENT(LAST)	AV(F)	NO(B1)	NO

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES	YES	YES	YES	YES	YES	YES
YES	YES	YES	YES	YES	YES	YES
YES	YES	YES	YES	YES	YES	YES
YES(D1)	YES(E1)	YES	NO	YES	NO	NO
YES	YES	YES	YES	YES(H1)	NO	NO

(D1) Available through IDS and COBOL.

(E1) Maximum of four.

(H1) The order must be dependent on values for specified elements and not dependent on order in storage.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.210 GENERAL RETRIEVAL CAPABILITIES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Temporary File - A temporary file for additional processing is a file which is a subset of and has the same logical organization as the file from which it was produced. This temporary file may be used as an input file in another statement or output presentation.	YES	NO(B1)	YES
Number of Batched Queries - The capability of processing more than one retrieval specification (query) in a single pass of the file. The object is to amortize file search time across a number of queries.	NO	NA	NF
Indirect Query - A process in which retrieved data element values from one or more logical expressions can be applied as literal input values in another logical expression of the same operational retrieval statement.	YES	NO	NF
2.220 PROCEDURAL LANGUAGE			
Repetitive (e.g., DO Statement) - Whether a system provides a direct looping logic capability such that retrieval statements may be executed a user specified number of times.	NA(A1)	YES	NO
Any Execution Sequence - Implies the capability to execute any sequence of retrieval statements not necessarily in serial order, i.e., allowing unconditional branching (e.g., GO TO statement).	NA(A1)	YES	YES(C1)

(A1) Function of procedural language (e.g., COBOL), not of DM-1 language.
(B1) Available through COBOL procedure.
(C1) Control cannot be passed (i.e., by GO TO) out of subfunction level.
(D1) Available through IDS and COBOL.
(E1) Limited by memory size and complexity of queries.
(F1) Requires user intervention to combine individual logic statements into single subprocedures.
(G1) QDF (Qualified Data File).

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U. S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES(G1)	YES	YES	YES
NA	(E1)	NO(F1)	(G2)	(H1)	(I1)	NA
YES(D1)	YES	YES	NO	YES	NO	NO
YES(D1)	YES	YES	NO	NO(H2)	NO	YES
YES(D1)	NO	YES	NO	NO(H2)	NO	YES

(G2) Dynamic Parameter fixed at execution time, but may change as a generation function.

(H1) The user has the effect of batching queries by being able to specify many queries in one run; but because of the file structure, processing these in one pass is meaningless.

(H2) Available by writing a JOVIAL procedure.

(I1) The number of queries that can be batched is a function of the size of core and length of query.

*U. S. ARMY SYSTEM DEVELOPED BY CDC

2.220 PROCEDURAL LANGUAGE (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Conditional Execution (e.g., IF Statement) In some circumstances, the user may want certain retrieval statements to be executed only under a specific condition.	NA(A1)	YES	(C1)
EXAMPLE: BEGIN IF "AGE" GTE (30) GO TO A. I = I + 1 GO TO BEGIN. A IF "TITLE" EQ (STAFF) . . . etc.			
2.230 LANGUAGE INTERFACE			
Other System Processors - Allows the user to directly call on another system processor and use its language to query a file in the physical logical format established by the DMS. The specific languages considered are:		(B1)	(C2)
- Assembly Language	YES(A2)		
- PL/I	YES(A2)		
- COBOL	YES(A2)		
- FORTRAN	NO		
- ALGOL	NO		
- JOVIAL	NO		
Library Subroutine Links - Whether the system provides the capability of calling a subroutine from either the DMS or operating system library as part of the retrieval specification.	YES	YES	YES

(A1) Function of procedural language (e.g., COBOL), not of DM-1 language.

(A2) Achieved in processing programs written in PL/I, COBOL, or BAL by special interface with DM-1 language .

(B1) FORTRAN and COBOL.

(C1) Control cannot be passed (i. e, by GO TO) out of subfunction level.

(C2) If available with the operating system under which COGENT III is run.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NF	YES	NO	NO(H1)	NO	YES
	(E1)	NO(F1)	NO	(H1)	NO	NO
YES NO YES(D1) NO NO NO						
NA	YES	YES	YES(G1)	(H1)	NO	YES

(D1) Available through IDS and COBOL.

(E1) Capability is provided to use routines written in any language which fits in the IBM operating system.

(F1) GIS files are processable by BAL, COBOL, and PL/I programs, assuming reasonable care in file definition.

(G1) Geographical operators.

(H1) Available by writing a JOVIAL procedure.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

2.240 ANCILLARY LANGUAGE CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Temporary Hold File - A temporary hold file is a file of data values retrieved from the system file which is not in the same logical or physical structure as the system file. A primary use of a "hold" file is to produce additional copies of previous outputs or reports.	YES	NO(B1)	YES
Prestore Queries - Allows the user to store a query on a system library and call the query at execution time. This capability is especially useful for frequently run queries.	AV(F)	NO(B1)	YES
Modify Prestored Queries			
- Stored Library Form - Allows the user to change the source form of the query as it is stored on the library. This is a permanent change.	AV(F)	NO	NF
- Temporary Modification (1 Run) - Allows the user to change the prestored query at execution time; the query as it is on the library is unchanged.	AV(F)	NO	NF
Parameterized Queries - In some systems, it is possible to store skeletal queries on a library; the operands, output, and sort statements, for example, may be omitted. This capability permits the user to vary these parameters at execution time depending on the specific query to be executed.	AV(F)	NO(B1)	YES

(B1) Available through COBOL procedure.

(D1) Available through IDS, COBOL, and FILE MANAGER.
(D2) Available through IDS, COBOL, and LIBRARIAN.
(D3) Available through IDS and COBOL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	(G1)	NO	NO	YES
YES(D2)	YES	YES	YES	YES	NO	YES
NO	YES	YES	NO(G2)	YES	NO	YES
NO	YES	(F1)	NO(G2)	YES	NO	YES
YES(D3)	NO	YES(F2)	NO(G2)	YES	NO	YES

(F1) If modification of the query is planned, parameters can be accessed at execution time.

(F2) Can parameterize qualification values, but not statements (i. e., SORT) or operators.

(G1) QDF (Qualified Data File).

(G2) Capabilities will not be available in Phase I; they are planned for Phase II.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.000 MAINTENANCE

File maintenance is the process by which new files can be added to the DMS data base and already existing files changed physically or logically. It includes:

(a) File generation (the initial step in the cycle), which consists of two processes: describing the logical format of the file and mapping the input data into this format;

(b) File updating, the process of changing data in a file already in existence, a process that affects the physical structure or content of the file onl;r; and

(c) File creation, the process of creating a new system file from a previously generated system file, a process that may change both the logical and physical structure of the file.

The data used to maintain a file are referred to as source data, or transaction data, and may be either structured or unstructured. Structured data have a format that can be defined (by a format table which specifies which fields correspond to the defined data elements). An example of unstructured data is a transaction that includes both the name of the data element and its corresponding value, and which therefore requires no format table.

Transaction data media may be cards, magnetic tape, direct-access devices, typewriters, remote consoles, etc. The process of mapping transaction data from medium to file can be provided with varying degrees of flexibility. In some systems, this mapping is under complete control of the DMS, while in others the user has extensive capa' ilities. In the systems where the user has some control, the control is generally exercised by procedural types of statements. These are similar to compiler statements in that the user can specify what is to be done for each particular case. The basic capabilities include:

(a) comparisons of values
(b) addition, subtraction, multiplication, and division of values

(c) looping logic
(d) error conditions and actions
(e) replacing one value with another.

Since not all transaction data are in a form that is suitable for the file, most systems provide the user with some data validation and editing capabilities. Validation is the process of checking the contents of data, e.g., a check for only numeric characters in a social security number. Editing is the process of changing the form of the data, e.g., eliminating the hyphens in the social security number.

File generation and updating are reasonably straightforward. File creation, on the other hand, is more complex. File creation is the process of restructuring the file logically as well as physically. During file creation at least two file formats are required, one for the old (source) file and one for the new file which is to be created. The transaction data and the data from the old file are selectively mapped into the new file based upon the user-specified procedure statements.

3.100 General Capability

This is the general capability that the system provides for making changes to already existing files. This is the process in which transactions are applied to the files in the manner dictated by the user-specified procedures.

3.110 Multi-File

Multi-file operations involve more than one file of a particular type. The files may have different physical and logical formats. The two basic multi-file operations are:

(a) one source file, several destination files
(b) several source files, one destination file.

3.120 On-Line Maintenance

On-line maintenance is the entering of transaction data, procedure statements, data validation, and editing conditions from an on-line device at the time of the maintenance run.

3.130 Logical Maintenance

Logical maintenance activity is conditional execution of maintenance functions based on logical (Boolean) criteria. The logical maintenance capabilities included are:

(a) temporarily overriding defined attributes of data

(b) simultaneous revision of many instances of a data element

(c) arithmetic operations as data-value revision operators.

3.140 Arithmetic Maintenance

Arithmetic maintenance capabilities are those provided for the addition, subtraction, multiplication, division, exponentiation, square root, trigonometric functions, etc., of data element values, constants, and literals. The topics considered here are the following:

(a) the use of literals in arithmetic operations

(b) the use of data element values in arithmetic computation of the values of other data elements

(c) the specific arithmetic operators which are available.

3.150 Specific Capabilities

The specific capabilities considered here are those which can be thought of as maintenance in the large, as opposed to the tactical capabilities described above. These include the following:

(a) data file modification, deletion, addition of entries

(b) multiple simultaneous data-file operations

(c) aggregation of batched data-file operation requests for later execution

(d) data-directed specifications for data-file operations

(e) accumulation of histories of data-file operation runs.

3.160 Accessibility and Modification of File Maintenance Procedures

File maintenance procedures may be used frequently and must be modified from time to time because of changing update requirements or changing transaction data formats. This capability is analogous to that described above in 2.240. It includes the following features:

(a) creation and storage of file-maintenance procedures

(b) modification of file-maintenance procedures

(c) parameterized execution of file-maintenance procedures.

3.170 Error Procedures

The flexibility provided to the user in determining the procedures to be followed when error conditions are encountered includes:

(a) pre-set user specification of error procedures

(b) dynamic (conversational) user specification of error procedures

(c) capability to retrieve all erroneous data records as an entity, such retrieval being either predetermined or dynamically determined.

3.110 MULTI-FILE	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Update Several Files - One transaction file can be used to update more than one data file during one update job. Simultaneously means that there is only one pass of the transaction file and each system file during the job.	AV(F)(A1)	YES(B1)	YES
Use Several Transaction Files - Two or more input transaction files can update a single-system data file without requiring either a second pass of the data file or previous merging of the input transaction files.	AV(F)(A1	NO(B2)	YES
3.120 ON-LINE MAINTENANCE			
Whether the user can specify whether to, and how to, perform maintenance operations by using an on-line (conversational) communication device.	AV(F)(A1)	NO(B2)	YES
3.130 LOGICAL MAINTENANCE			
Override Data Definition - The capability to nullify the previously established data-validation conditions or the data definition. (Data definition usually occurs at file-generation time: it is the specification of the data element name, the number of characters in its value, the types of validity checking which the value must undergo, etc.).	AV(F)(A1)	NO(B2)	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Capability can be added to File Maintenance programs.

(B2) Available through COBOL procedure.

(D1) Available through IDS and COBOL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NF	YES(F1)	NO(G1)	NO	NO	NO(J1)
YES(D1)	YES	YES(F1)	YES	YES	NO	NO(J1)
NO	NO	YES(F2)	NO(G1)	YES	NO	NO(J1)
NO	NO	YES	YES	YES	NO	NO(J1)

(F1) Modify mode only. Changes to modification of existing fields (i.e., no new segments or records). Up to 16 files, including both transaction and master.

(F2) System primarily designed for batch operation.

(G1) Available in Phase II.

(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.130 LOGICAL MAINTENANCE (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Revise All Instances - Whether the user can specify how to revise all occurrences of a specific data element in a file based upon a specified condition; e.g., for all cities with a population less than 500, replace CITY with the word TOWN.	AV(F)(A1)	NO(B1)	YES
Perform Arithmetic Function - Whether arithmetic operations can be performed on all instances of a data element in a file that meet a user-specified condition; e.g., for cities with population of greater than 475,000 add 25,000 to the population.	AV(F)(A1)	NO(B1)	YES
3.140 ARITHMETIC MAINTENANCE			
Literals - This is the use of a literal in a maintenance procedure statement; e.g., add (100,000) to population where 100,000 is a literal and population is a data element name (which has a value in a specific instance).	AV(F)(A1)	YES	YES
Compute New Data Value - The capability to compute a new data value from other data values; e.g., compute state population by adding the populations of all cities in the state.	AV(F)	NO(B1)	YES
Arithmetic Operators - The ability to use the arithmetic operators =, +, -, /, EXP, in maintenance transactions.	NF	YES	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	NO	NO(J1)
YES(D1)	YES	YES	YES	YES	NO	NO(J1)
YES(D1)	YES	YES	YES	YES	YES	NO(J1)
YES(D1)	YES	YES	YES	YES	NO	NO(J1)
YES(D1)	YES	YES	YES	YES	+, -	NO(J1)

(D1) Available through IDS and COBOL.

(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.150 SPECIFIC CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Add - The addition of physical values, sets of values, or entries to a file that has previously defined their logical counterparts of data element, group, or logical entry.	YES	ENTRY ONLY	YES
Delete - The deletion of physical values, sets of values, or entries to a file that has previously defined their logical counterparts of data element, group, or logical entry.	YES	ENTRY ONLY	YES
Modify - The modification of physical values, sets of values, or entries to a file that has previously defined their logical counterparts of data element, group, or logical entry.	YES	YES	YES
Hold and Batch - The ability to aggregate update transactions for a different maintenance run.	AV(F)(A1)	AV(O)	YES
Data Directed - Whether both the name of the data element and its corresponding value can appear on the update transaction, thus eliminating any transaction format table.	AV(F)(A1)	YES	YES
List Changes - Whether a listing indicating all additions, deletions, and modifications made to a file during a maintenance run can be produced either automatically or upon user request.	AV(F)(A1)	YES	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES	YES	YES	YES	YES	ENTRY ONLY	NO(J1)
YES	YES	YES	YES	YES	ENTRY ONLY	NO(J1)
YES	YES	YES	YES	YES	DATA ELEMENT ONLY	
YES(D1)	YES	YES(F1)	NO	YES	YES(I1)	NO
YES(D1)	NO	NO	YES	YES	YES	NO
YES(D1)	YES	YES	YES	NO	YES	YES

(D1) Available through IDS and COBOL.
(F1) Transaction batching can be accomplished with a prestored update procedure. Each set of transactions will cause an independent execution of this procedure.
(I1) Limited to card decks or card image tape file.
(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.150 SPECIFIC CAPABILITIES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Query While Updating - Whether the system permits two independent accesses to the same file at the same time (not necessarily simultaneously); e.g., if User A is updating a file, whether it is possible for User B to query the same file at the same time. In such a situation, User B might not retrieve the most current information because the update may not be completed. Sometimes it is important to obtain an answer even if it is not the most current one.	(A1)	NO(B1)	NF
3.160 ACCESSIBILITY AND MODIFICATION OF FILE MAINTENANCE PROCEDURES			
Prestore - Procedures for file maintenance may be lengthy and used quite frequently. Some DMS's provide for storing of these procedures in the source form, while others provide for the storage in the object form, usually on a system library. Procedures stored in the source form must be compiled each time they are executed (unless they are executed interpretively), but are easy to change either permanently or temporarily. Object-form storage eliminates the compile process at each execution, but any required changes are considerably more difficult.	AV(F)(A2)	YES	YES

(A1) The data base is divided into update units. Any number of queries can use the same update unit, but an update run has exclusive access to an update unit until the update unit is released.

(A2) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	NO	NO	NO	NO	YES
YES(D2)	YES	YES	YES	YES	YES	YES

(D1) Available through IDS and OS.
(D2) Available through LIBRARIAN.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.160 ACCESSIBILITY AND MODIFICATION OF FILE MAINTENANCE PROCEDURES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Modify - The capability to change pre-stored maintenance procedures, either permanently or temporarily:			
- Stored Library Form - The procedure as stored on the library is modified and the change is permanent.	NO	NO	YES
- Temporary (1 Run) - This modification is made at run time and is effective only for the current run.	AV(F)(A1)	NO	YES
Parameterize - Whether parameterized procedures can be prestored and the parameters entered at execution time.	AV(F)(A1)	YES	YES
3.170 MAINTENANCE ERROR PROCEDURES			
User Specification - Whether the user can specify what actions are to occur (terminate run, print error comment, etc.) if there are errors in the transaction data, procedure statements, etc.	NF	YES	YES
On-Line Correction - Whether the user can correct any data element value errors on-line during the execution of the maintenance run.	NF	NO(B1)	YES
Print Error Records - Whether the user can request that a summary of all designated error records be provided at the completion of the maintenance run.	AV(F)(A1)	NO	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	YES	NO
NO	YES	NO	YES	YES	NO	NO
YES(D1)	NO	NO	YES	YES	NO	NO
YES	YES	YES	YES	NO	NO	NO
(D2)	NO	NO	NO(G1)	YES	NO	NO
YES(D3)	YES	YES	YES	YES	NO	NO

(D1) Available through LIBRARIAN.
(D2) Available through IDS and COBOL.
(D3) Available through COBOL.

(G1) Available in Phase II.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.200 Input Processing/Data Validation

One of the major problems in maintaining data bases is the quality and validity of data. Many systems must depend on preprocessors because of lack of data-validation procedures.

3.210 General Data Validation

These are the capabilities which are provided for ensuring that the transaction data are in the correct form to be entered into the system. At the time that a data base is defined the user can specify the types of validation which are to be performed on each data element, and the user can also specify data values against which data entering the system can later be compared. At data-transaction time the DMS compares each entering data value against the pre-defined validation criteria, and accepts only those entering data values which satisfy the criteria. The user can usually modify the criteria, either permanently or temporarily, at times other than data-base definition time. The different types of validation which a DMS may provide are described here.

3.220 Input Edit

These indicate whether the system can change the form of transaction input data values prior to inserting them in the file. Again, the user can define the desired editing at the time the data base is being defined, and can usually modify the editing specifications at later times. Editing specification consists of the specification of the general type of editing as well as specification of literal values to be used in the editing operation. The general types of editing which a DMS may provide are de -ribed here.

3.210 GENERAL VALIDATION	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Minimum/Maximum - A check applied to numerical data values to ensure that the input data value is either greater than a user-specified value or that it is less than another user-specified value; e.g., population greater than 500.	AV(F)(A1)	NO(B1)	YES
Range - A check applied to numerical data values to ensure that the input data value lies within two given values. The limit values may be either exclusive or inclusive. For example, population greater than 10,000, but less than 25,000.	AV(F)(A1)	NO(B1)	YES
Specific Characters - A check applied to specific characters of a data value; e.g., numeric elements could be validated to ensure that there are no alpha or special characters or alpha elements could be validated to ensure no numeric characters. The population 5R6 is invalid.	AV(F)(A1)	NO(B1)	YES
Sequence or Identification - Some DMS's require that the input transaction data be sequenced in a specific order. Failure to so sequence the data results in a termination of the maintenance run or an invalid maintenance run. Such sequence checks may reveal that transactions are out of sequence or missing, or that the entire transaction file is invalid.	AV(F)(A1)	YES	YES
Cross Comparison - The ability to accept values of one data element on an input transaction based upon the specific value of another data element in the transaction.	AV(F)(A1)	NO	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	YES	YES(F1)	YES	YES	NO	(J1)
YES(D1)	YES	YES(F2)	YES	YES	NO	(J1)
(D1)	YES	YES	YES	YES	NO	NO
NO	YES	YES	YES	NO	YES	(J1)
NO	YES	YES(F3)	YES	NO	NO	(J1)

(D1) Where data declaration of master selection or duplicate detail criteria are applied.

(F1) Using RANGE statement and system upper- and lower-limit values.
(F2) Multiple ranges are permitted.
(F3) Using specific logic statements to perform a comparison.

(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.210 GENERAL VALIDATION (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
General Format - The system accepts or rejects a transaction based upon the validity of the predefined format of that transaction; e.g., all-numeric fields or 70-character records.	AV(F)(A1)	YES	YES
3.220 EDIT SPECIFICATIONS			
Modify Data Element Size - Whether the system provides the capability to add or delete characters from the input data value; e.g., deletion of the abbreviation CO. from the input data value CALHOUN CO.	AV(F)	NO	YES
Automatic Truncation or Padding - Whether the system can truncate or pad the input data values with zeros, blanks, etc.; e.g., pad 11673 to become 00011673 or truncate 3.1414 to 3.14.	YES	YES	YES
Add Information - Whether input data element values can be augmented with other data; e.g., add N to the input data value LATITUDE +43 20, yielding N43 20.	AV(F)	NO	YES
Decode - Whether the system can decode input data values into natural formats; e.g., US becomes UNITED STATES OF AMERICA.	NF	NO(B1)	YES
Encode - Whether the system can encode input data element values (the converse of the above); e.g., UNITED STATES OF AMERICA becomes US.	NF	NO(B1)	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	YES	YES	YES	YES	YES	(J1)
NO	YES	YES(F1)	YES	YES(H1)	NO	(J1)
(D1)	YES	YES	YES	YES(H1)	NO	(J1)
NO	NO	YES(F2)	YES	YES(H1)	NO	(J1)
NO	(E1)	YES	YES	YES	NO	(J1)
NO	(E1)	YES	YES	YES(H1)	NO	(J1)

(D1) Available through COBOL.
(E1) Available as an additional option to the basic MARK IV system.
(F1) Using procedural language; i.e., by the verb CONCATENATION.
(F2) Can only be performed using edit subroutine.
(H1) Accomplished via a generalized system preprocessor.
(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.300 File Generation

File generation is the capability to define a logical file organization and to map the input transaction data into a physical file with the specified logical organization. This is the process by which a file and its data element values are entered into the system.

3.310 Intra-System

This is the capability that the system provides for accepting transaction data which have been generated by various other system software processors (within the same operating system) and which have been stored on various physical media. The entries indicate whether the system can use files generated in COBOL, ALGOL, PL/I, FORTRAN, etc., as transaction data, on what media these files can be stored (tape, disc, etc.), and whether transaction data can be accepted on-line from a remote console.

3.320 Inter-System

This is the capability that the system provides for accepting transaction data files generated on other computers or under different operating systems. Such files are referred to as foreign files. The entries indicate whether the system can use foreign files at all, and, if so, on what media these files can be stored (tape, disc, etc.).

3.330 Physical Format of Data to be Generated

This indicates whether the system allows the format of input data to be:

(a) the same for all data
(b) one of several formats for each data file
(c) self-defined at data input time.

3.340 Transformation Control of Input Data

This is the capability that the system provides for controlling the mapping of data from the input transaction to the file being generated. This can be a physical characteristic which indicates the end of a repeating group, entry or transaction.

3.310 INTRA-SYSTEM	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Physical Media - These are the system input devices from which the DMS can accept transaction data.	NA(A1)		
- Cards		YES	YES
- Disc		YES	YES
- Mag Tape		YES	YES
- Other		PAPER TAPE	NO
Remote Devices - This indicates whether the DMS can accept transaction data from remote devices.	NA(A1)	YES	YES
Other System Processors - These indicate whether the DMS can accept transaction files produced by other system processors within the operating system under which the DMS operates.	AV(F)(A2)	(B1)	YES(C1)
- COBOL			
- FORTRAN			
- PL/I			
- ALGOL			
- Other			
3.320 INTER-SYSTEM			
Foreign Files	NO	YES	YES
Physical Media	NA(A1)		
- Mag Tape		YES	YES
- Disc Packs		NO	YES
- Other		NO	YES

(A1) DM-1 is indifferent to input source.
(A2) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.
(B1) COBOL, FCRTRAN, ALGOL.
(C1) If the files are processable by COBOL.
(D1) Available through COBOL and FILE MANAGER.
(D2) Available through TERMINAL SUPERVISOR.
(D3) Assembly language.
(E1) If physical device is compatible with the IBM 360.
(F1) Actual input assumes file organizations which can be described to GIS. Not all PL/I and FORTRAN formats can be so described.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	YES	YES
YES	YES	YES	YES	YES	NO	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	DRUM	NA	LIM(O)	NO	NO	NO
YES(D2)	NO	YES	NO(G1)	YES	NO	NO
			(G2)		YES(I1)	
YES		YES		YES(H1)		YES
NO		YES(F1)		YES(H1)		YES
NO		YES(F1)		YES(H1)		NO
NO		YES(F1)		YES(H1)		YES
YES(D3)		NA		YES(H1)		NO
NO	YES(E1)	NO(F2)	(G2)	YES(H2)	YES(I1)	YES
YES(D1)	YES(E1)	NO(F2)	(G2)	YES(H2)	YES(I1)	YES
YES(D1)	YES(E1)	NO(F2)	(G2)	YES(H2)	NO	YES
YES(D1)	YES(E1)	NO(F2)	(G2)	YES(H2)	NO	NO

(F2) If OS/360 and GIS file descriptions can handle the organization of the data, then the system can process it.
(G1) Available in Phase II.
(G2) Limited only by the capability to describe the files in the Transaction Descriptor Deck (TDD).
(H1) Any processor that can produce character-type records.
(H2) Accomplished by a generalized system processor.
(I1) Any fixed-formatted BCD tape file (729II or IV).

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.330 PHYSICAL FORMAT OF THE INPUT DATA TO BE GENERATED	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Specific Format - An explicit one-to-one correspondence between the physical sequence of the input data and the logical organization of the file to be generated; e.g., the physical sequential organization (see 1.220) of hierarchical files on magnetic tape. This limits the flexibility for formatting the input data.	AV(F)(A1)	NO	NF
Several Formats - Whether the system allows different physical formats for the input transaction data during file generation.	AV(F)(A1)	YES	NF
Any Format - Whether the input data may be in any physical format that can be defined to the DMS. This format need not be in direct one-to-one correspondence with the logical organization of the files to be generated.	AV(F)(A1)	YES	NF

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO	NO	NO	NO	NO	NO
YES	YES	YES	YES	YES(H1)	YES	YES
YES	YES(E1)	YES	NO	YES(H1)	(I1)	YES

(E1) Control information in the record identifies one of several record formats or processing definitions.

(H1) Accomplished by a generalized system processor.

(I1) Any format that can be defined using cards/card image.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.340 TRANSFORMATION CONTROL	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Embedded Control Fields - A standard control field applicable to all transactions (in all applications); e.g., the word END in a transaction would always be interpreted by the system as the end of that transaction.	AV(F)(A1)	YES	NF
Special Character(s - The termination symbol might be one or a series of special characters; e.g., &, +++, etc.	AV(F)(A1)	NO	NF
Physical Media - A termination symbol independent of the actual transaction data that is hardware detectable; e.g., recording gap, end-of-file mark.	AV(F)(A1)	YES	NF
Input Data - This is a user-specified termination symbol which appears in the input transaction data, and which may invoke a special translation procedure.	AV(F)(A1)	YES	NF

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	(F1)	YES	YES(H1)	NO	YES
YES(D1)	YES	(F1)	YES	YES(H1)	NO	YES
YES(D1)	YES	(F1)	NO	YES(H1)	YES	YES
YES(D1)	YES(E1)	(F1)	YES	YES(H1)	YES	NO

(D1) Available through COBOL.

(E1) Control information in the record identifies one of several record formats or processing definitions.

(F1) GIS file generation is performed by describing the source data as a GIS file, and then reformatting if required.

(H1) Accomplished by a generalized system processor.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.400 File Creation

File creation is the process by which a new file is created from one or more already existing system files. The created file may have a different logical and physical structure. Files may be restructured to eliminate or add data elements, to redefine the size of data elements, to reorder groups and instances of groups, to repack data, to subset the file for another user, etc.

3.410 Specific Capabilities

Whenever a new file is created from another, data element values must be transferred from file to file. The data elements to be transferred can be selected on a conditional basis, and their values can be validated and edited as in 3.210 and 3.220 above. Data definitions can also be temporarily overridden, as in 3.130 above.

3.420 System Files

This is the general capability to create a new system file from an old. A new file can be created by restructuring an old file logically or physically, by sorting the entries or subentries of an old file, by merging several old files together, or by taking a subset of an old file.

3.430 Data Preparation--Inter-/Intra-System

This is the capability that the DMS provides for preparing data files which can later be processed by other DMS's or by other subprocessors of the generating DMS. These prepared data files can be characterized in terms of their logical and physical formats, of the storage media on which they reside, and of the system processors or subprocessors for which they have been prepared.

3.410 SPECIFIC CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
<u>Conditional Selection</u> - The capability to select desired data elements from a source file(s) for a newly created file based on a logical criteria.	YES	NO(B1)	YES
<u>Validity Checking (see 3.200)</u> - These are the same as those in 3.200, but apply to both transaction data and the data elements in the source file.	AV(F)(A1)	YES	YES
<u>Override Data Definition</u> - Whether the system permits the user to nullify predefined encoding/decoding options, checks, etc., for data element values in both the source file and the new file.	AV(F)(A1)	NO	NO
3.420 SYSTEM FILES			
<u>Restructure File</u>			
- <u>Logically</u> - The created file has a different logical structure than the source file. This implies that the file definition has been changed, e.g., by elimination or addition of data elements or groups, or by different ordering of data elements.	AV(F)(A2)	NO(B1)	YES
- <u>Physically</u> - The created file has a different physical structure than the source file. The file definition is the same. An example of physical restructuring is repacking a file to reallocate storage.	AV(F)(A2)	NO(B1)	YES
<u>Reorder Group Instances</u> - The ability to reorder the physical instances of a group based on the value of a data element in the group.	AV(F)(A2)	NO(B1)	NO

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.
(A2) Available in 2nd quarter, 1969.
(B1) Available through COBOL procedure.
(F1) New files, but not source files.
(F2) Automatic (via new data description table) for subordinate groups. Two-step procedure (hold file, and sort) for master group.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NF	YES	YES	YES	YES	YES	YES
NO	YES	YES	YES(G1)	YES	NO	NO(J1)
NO	YES	(F1)	NA	YES	NO	NO
NF	YES	YES	YES(G2)	AV(F)	YES	YES
NF	YES	YES	YES(G2)	AV(F)	YES(I1)	YES
NF	NO	YES(F2)	NO	AV(F)	NA	NF

(G1) Analyst has capability to override.
(G2) File revision module.

(I1) Reorganize field positions.

(J1) RAPID uses a special-language maintenance program.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.420 SYSTEM FILES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Multi-File - (see 3.110)			
- Create - From one system source file, create two or more new files.	AV(F)(A1)	NO(B1)	YES
- Merge - Combine two or more similarly ordered system files into a single system file.	AV(F)(A1)	NO	YES
3.430 DATA PREPARATION			
3.431 INTRA-SYSTEM			
This is the capability to prepare data files which are processable by other language processors which are under the same operating system as the DMS.			
Other Files - The capability to create files that can be used as input data files for other programs written in FORTRAN, COBOL, PL/I, etc. The DMS provides all the file-processing control words and control blocks necessary for the languages to process those files.		(B2)	(C1)
- FORTRAN	NF		
- COBOL	YES		
- ASSEMBLY LANGUAGE	YES		
- PL/I	YES		
- ALGOL	NF		
- JOVIAL	YES		
Other Internal Data Structures - Internal data structures are the data-organization mechanism of other language processors. Examples of these are FORTRAN arrays or JOVIAL tables.	NO	YES	(C1)

(A1) Available in 2nd quarter, 1969.
(B1) Available through COBOL procedure.
(B2) COBOL, FORTRAN, ALGOL.
(C1) If available through a COBOL procedure.
(D1) Available through COBOL.
(E1) Maximum of four.
(E2) Nearly any output format accessible by OS and DOS, QISAM and QSAM.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	YES	NO	NO	YES	NO	YES
NO	YES(E1)	YES(F1)	YES	AV(F)	NO	YES
	(E2)		(G1)			
NO		YES(F2)		NO	YES(I1)	YES
YES		YES(F2)		NO	YES(I1)	YES
YES		YES(F2)		NO	YES(I1)	NO
NO		YES(F2)		NO	YES(I1)	NO
NO		YES(F2)		NO	YES(I1)	YES
-		-		YES	-	NO
YES(D1)	NF	NO	NO	NF	NO	NO

(F1) Requires two-step procedure using HOLD file as subsequent source file.

(F2) Actual input assumes file organizations which can be described to GIS. Not all PL/I and FORTRAN formats can be so described.

(G1) Assembly language.

(I1) Any fixed-formatted BCD tape file (729II or IV).

*U.S. ARMY SYSTEM DEVELOPED BY CDC

3.432 INTER-SYSTEM	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
This is the capability that the DMS provides for creating files using a variety of media and for defining files which are to be used in other systems.			
Physical Media			
- Unit Record	NA(A1)	YES	YES
- Tape	NA(A1)	YES	YES
- Disc Pack	NA(A1)	NO	YES
- Other	NA(A1)	NO	-
Data Descriptions - The data specification necessary for the new system file.			
- Target Data Management System	AV(F)	NO	YES
- Self-describing Files	AV(F)	NO	YES
Foreign Files	AV(F)	YES	(C1)

(A1) DM-1 is indifferent to device type.

(C1) If available through a COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	NF	YES	YES
YES(D1)	YES	LIM(O)	YES	NF	YES	YES
YES(D1)	YES	LIM(O)	YES	NF	NO	YES
NO	(E1)	LIM(O)	LIM(O)	NF	NO	NO
YES	NO	NO	NO	NF	NO	NO
NO	NO	NO	NO	NF	NO	NO
NO	NO	NO	NO	NF	YES(I1)	YES

(D1) Available through COBOL and FILE MANAGER.

(E1) Data cell, drum.

(I1) Reorganize field positions.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.000 OUTPUT PRESENTATION

Output presentation is the process by which the system delivers the results of user queries through the system output devices. The presentation of the output is the last stage of file processing. It is at this point that the user gets the opportunity to observe the results of his query processes. Output presentation can be controlled either through functions or user requests.

The output presentation may be indicative of the performance of the DMS. Analysis of the output can yield useful information concerning the performance of the other parts of the DMS and can be a useful aid in debugging a system.

4.100 Standard System-Supplied Formats

"Standard" system-supplied formats are those which are an integral part of the system and can be provided for the user either automatically or upon specific request.

4.110 General Capabilities

The following types of capability are sometimes available in the system as part of standard system formats:

(a) general output-formatting capabilities

(b) output contents and value-transformation capabilities

(c) special output capabilities.

4.120 Output Page Headers

This is the capability to provide various types of information at or near the top of a page of output. The information include titles, security classification, dates, page numbers, etc.

4.130 Trailer Information

This is the capability to provide various types of information at or near the bottom of a page of output.

4.110 STANDARD REPORTS--GENERAL CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Column Width - Whether the system can present outputs in balanced columns which are automatically adjusted to the number of characters in either the data element name or value. EXAMPLE: CITY POPULATION STATE Baltimore 670,000 Maryland Damascus 980,000 Maryland	AV(F)(A1)	YES	NF
Output Line - Whether the system adjusts the number of characters in a line of output to correspond to the maximum number of characters per line that can be printed by the output device, e.g., print 64 characters of teletypewriter message and automatically execute a carriage return/line feed before printing the next character.	AV(F)(A1)	NO	NF
Position on a Page - Whether the system can position data element names and their corresponding values in a meaningful way. EXAMPLE: CITY STATE Boston Massachusetts Rome New York	AV(F)(A1)	NO	YES
Report Title - Whether the system can select and print a report title or heading provided by the user.	AV(F)(A1)	NO	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/1, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D2)	YES	YES(F1)	YES	YES	YES	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	YES(E1)	YES	YES	YES	YES	YES

(D1) Available through COBOL.
(D2) Available through TERMINAL SUPERVISOR.

(E1) Title restricted only by physical page size.

(F1) System upper limit of 128 bytes.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.110 STANDARD REPORTS--GENERAL CAPABILITIES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Editing and Decoding - Whether the system provides automatic editing functions such as suppression of leading zeros and insertion of algebraic signs, dollar signs, and punctuation. Decoding indicates the ability of the system to supply actual values for coded data-element values which are stored in coded form in the data base. These functions are usually defined at file-generation time and stored in a corresponding data-description table. EXAMPLE: If the value "United States of America" is stored in the data base as US, the system will produce as output "United States of America" and not US.	AV(F)(A1)	NO	YES
Different Reports - Whether the system can generate different report formats from one retrieval statement.	NO	YES	YES
Multiple Copies - Whether the system can provide more than one original copy of the same report as distinguished from multiply printed output.	NO	YES	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	EDIT	YES
(D2)	YES	YES	YES	NO	YES	YES
(D3)	YES	NO	YES	NO	YES	YES

(D1) Available through COBOL.
(D2) Available through IDS and COBOL.
(D3) Available through FILE MANAGER.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.110 STANDARD REPORTS--GENERAL CAPABILITIES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Logical File Structure - Whether the system can provide a list of the data elements which comprise a file formatted to indicate their logical relationships. EXAMPLE: NATION FILE NATION NAME PRESIDENT POLITICAL SUBDIVISION STATE NAME GOVERNOR CITIES CITY NAME POPULATION COUNTIES COUNTY NAME COUNTY SEAT	YES	NO	YES
Functional Formats - Whether the system can provide counts and totals of the retrieved data-element names and values. The accounting may be categorized in two ways: - Sums/Totals - A sum or a total of the data element values. EXAMPLE: CITY POPULATION Mobile 202, 779 Montgomery 134, 393 Little Rock 300, 000 637, 172	AV(F)(A1)	NA	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NA	YES	YES	YES	YES	YES(I1)	NO
(D1)	YES	YES	YES	YES	YES	YES

(D1) Available through COBOL.

(I1) Print-out from dictionary generation run.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.110 STANDARD REPORTS--GENERAL CAPABILITIES (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- <u>Counts/Tallies</u> - A count or a tally of given data element occurrences. EXAMPLE: Using the above example, the NUMBER OF CITIES WITH POPULATION OVER 100,000: 3	AV(F)(A1)	NA	YES
4.120 OUTPUT PAGE HEADERS			
<u>Titles</u> - The ability of the system to provide titles for output report presentations.	AV(F)(A1)	NO	YES
<u>Date</u> - The ability of the system to provide and position the current date in each output presentation.	AV(F)(A1)	YES	YES
<u>Security Classification</u> - The ability of the system to provide the highest security classification of a file at the top and bottom of each output page.	AV(F)(A1)	NO	YES
<u>Page Numbering</u> - The ability of the system to sequentially number the pages of each output presentation.	AV(F)(A1)	YES	YES
<u>Table of Contents</u> - The ability of the system to produce a page listing the major report headings and their page numbers as the first page of an output presentation.	AV(F)(A1)	NO	NF
<u>Column Headings</u> - The ability of the system to provide the following types of column headings:			
- <u>Data Element Name</u> - The name of the data element appears at the head of a column of its values.	AV(F)(A1)	NO	NF

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
(D1)	YES	YES	YES	YES	YES	YES
(D1)	YES	YES	YES	YES	NO	YES
(D2)	YES	YES	YES	YES	NO	YES
NA	YES(E1)	NO	YES	LIM(O)	YES	YES
(D1)	YES	YES	YES	YES	YES	YES
NA	NO	NO	NO	NO	NO	NO
YES(D1)	YES	YES	YES	YES	YES	YES

(D1) Available through COBOL.
(D2) Available through COBOL and OS.
(E1) Security classification is part o' title.

*U.S ARMY SYSTEM DEVELOPED BY CDC

4.120 OUTPUT PAGE HEADERS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- Query Specified - A user-specified title in lieu of and/or in addition to the data element name.	AV(F)(A1)	NO	NF
- From Data Description - A columnar heading other than a data element name taken from the data description.	AV(F)(A1)	NO	YES
EXAMPLE: The data description for the data element POPU also indicates that the word POPULATION is to be supplied whenever values for POPU are presented as output.			
4.130 TRAILER INFORMATION			
This is the ability of the system to provide trailer information at the bottom of each output page of a report, usually including security classification and page number.	AV(F)(A1) AV(F)(A1)	NO NO	YES YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/1, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	NO	YES
NA	YES	YES	YES	YES	YES	YES
YES(D1)	YES	NO	YES	YES	YES	YES
YES(D1)	YES	NO	YES	YES	NO	YES

(D1) Available through COBOL.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.200 Reports--User Specified

User-specified reports are those which are designed and composed by an individual system user using the DMS report-creation capabilities. User-specified reports can be tailored to provide the exact output formats and contents desired.

4.210 General Capabilities

The types of capabilities provided by the DMS allow the user to specify the arrangement of, and value transformations for, data which is to be output. They also allow the user to cause the output to conform to preprinted or extraordinarily wide output forms.

4.220 Editing

The DMS will provide the capability to edit the physical appearance of data values which are to appear in user-specified reports. Such editing includes suppression of leading zeros, and insertion of algebraic signs (+, -), dollar signs, and punctuation symbols.

4.230 Pagination

Pagination control is the ability to control and number individual pages of an output report, either implicitly or explicitly. Explicit controls are user commands such as EJECT and SKIP. Implicit controls such as fixed page length and width are those which are built into the design of the system.

4.240 Output Media

User-specified reports can appear in a number of different output media. The following types of output media are considered here:

(a) on-line devices
(b) off-line devices
(c) audio devices.

4.210 GENERAL CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
The capabilities provided by the system tools for the user to specify report formats.			
Horizontal/Vertical Spacing - The ability of the system to effect any line and column positioning specified by the user.	AV(F)(A1)	YES	YES
Justification - The ability of the system to provide right or left justification of individual output fields as specified by the user.	AV(F)(A1)	YES	YES
Preprinted Forms - The ability of the system to position information to conform to the format of preprinted forms.	AV(F)(A1)	YES	YES
Override Decoding Transformations - The ability of the system to allow the user to obtain the actual value of the data element as it is stored in the data base, bypassing any decoding transformations ordinarily provided at data definition time.	AV(F)(A1)	NO	YES
Spread Sheet Output - The ability of the system to provide for a printout which exceeds the output line capacity of the device by placing the additional information on another sheet of output, the two sheets then joined along their widths for completeness.	NF	NO	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/1, COBOL, or BAL.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	NO	NO(F1)	YES	YES	NO	NO
YES(D1)	YES	YES LIM(O)	YES	YES	NO	YES
NA	YES	YES	YES	YES	NA	YES
NA	YES	NO	YES(G1)	YES	YES	YES

(D1) Available through COBOL.

(F1) Justification automatic (left for alphameric, right for numeric).

(G1) Requires another pass and a new RIT.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.220 EDITING	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
The following punctuation can be used by the system to edit data-element values as specified by the user. These functions are specified during output-format specification and are independent of data definition.			
Zero Suppress - The ability of the system to replace leading zeros with blanks.	AV(F)(A1)	YES	YES
Algebraic Signs - The ability of the system to insert a "+" or "-" sign depending on the sign of the data-element value.	AV(F)(A1)	YES	YES
Dollar Signs - The ability of the system to insert a dollar symbol ($) wherever specified, including before subtotals and totals.	AV(F)(A1)	YES	YES
Punctuation - The ability of the system to insert periods, commas, decimal points, slashes, and hyphens between data-element values or in specified positions.	AV(F)(A1)	(B1)	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) All except slashes.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES(G1)	YES	YES(I1)	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D1)	YES	YES	YES	YES(H1)	YES(I2)	YES

(D1) Available through COBOL.
(G1) Leading zeros can also be forced.
(H1) Between fields for all fields, as prefixes and suffixes in all fields, and decimal points at specified positions in numeric fields.
(I1) Function in dictionary.
(I2) Inserts ".", ",", "$", credit, or debit.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.230 PAGINATION	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Specify Starting Page - Whether the system allows the user to specify the starting page number and the increment to be used in determining the page number of succeeding pages. If no special pagination is specified, pages are numbered sequentially. EXAMPLE: First page is to be numbered 00010 and the increment number is 10, so that the second page is numbered 00020.	AV(F)(A1)	NO(B1)	YES
Break Page - Various methods can be used to stop printout on a given page and begin a new page independent of specified line counts:			
- Output Form - Page breaks are determined based on the format of the output form.	AV(F)(A1)	YES	NF
- Major Key - The system begins a new page based on a change in a major key. EXAMPLE: If the system is printing the cities in the US with population over 250,000 sorted alphabetically by states, the values for each state will start on a new page.	AV(F)(A1)	NO(B1)	YES
- Subtotal Value - If a specified subtotal value is reached, the system begins the next line of output on a new page.	AV(F)(A1)	NO(B1)	YES
- Count or Tally Value - If a specified count or tally value is reached, it is printed, and the next line of output appears on a new page.	AV(F)(A1)	NO(B1)	YES

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPIC
YES(D1)	(E1)	NO	YES	NO	NO	NO
YES(D1)	YES	YES	YES	YES	NO	YES
YES(D1)	YES	YES(F1)	YES	YES	YES	YES
YES(D1)	NO	YES(F1)	YES	NO	YES	NO
YES(D1)	NO	YES(F1)	YES	NO	YES	NO

(D1) Available through COBOL.

(E1) Starting page number, but not increment.

(F1) If requested in report procedure.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.230 PAGINATION (Continued)	AUERBACH	BURROUGHS	CSC
	DM 1	FORGE	COGENT III
Limit Output Volume - Whether the system can respond to user requests to limit the volume of output if it exceeds given thresholds:			
- Pages - Stop output after X pages	AV(F)(A1)	NO(B1)	YES
- Lines - Stop output after Y lines	AV(F)(A1)	NO(B1)	YES
- Data Element Values - Stop printout after Z data-element values have been printed.	AV(F)(A1)	NO(B1)	YES
4.240 OUTPUT MEDIA			
Three classes of output devices are generally used to present output information:			
On-Line - On-line operation provides rapid facilities to produce output for immediate analysis or use.			
- Typewriter - Keyboard input/output	NA(A2)	YES	YES(C1)
- Display - Video displays of text and vector output	NA(A2)	YES	YES(C1)
- Teletype - Standard ASR teletype input/output	NA(A2)	YES	YES(C1)
Off-Line - Off-line operation usually, but not necessarily, collects output for later presentation (e.g., as in a closed shop environment) in batch mode. The user's turn around time is generally much greater than that of on-line operation.			
- Tape	NA(A2)	YES	YES
- Disc	NA(A2)	YES	YES
- High Speed Printer	NA(A2)	YES	-
- Card Punch	NA(A2)	-	-

(A1) Capability to be available by 3rd quarter 1969. Meanwhile, available through programs written in PL/I, COBOL, or BAL.

(A2) DM-1 is indifferent to output device type.

(B1) Available through COBOL procedure.

(C1) If supported by the operating system.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES(F1)	YES	NO	NO	NO
YES(D1)	NO	YES(F1)	YES	NO	NO	YES
NO	YES	YES(F1)	YES	YES	NO	NO
YES	NO	YES	YES	YES	NA	YES
YES	NO	YES LIM(O)	YES	YES	NA	AV(O)
YES	NO	NO	NO(G1)	YES	NA	AV(O)
YES	YES	YES	YES	YES	YES	AV(O)
YES	YES	YES	YES	YES	NO	AV(O)
YES	YES	YES	YES	YES	YES	YES
-	YES		YES	YES	-	-

(D1) Available through COBOL.

(F1) If requested in report procedure.

(G1) Unless supported by operating system.

*U. S. ARMY SYSTEM DEVELOPED BY CDC

4.240 OUTPUT MEDIA (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Audio - Whether the hardware system provides for voice communication with the DMS in either of the following ways: - Spelled Voice - Words spelled 'or the user, e.g., bee, oh, ess, tee, oh, en. (Boston) - Spoken Voice - Words spoken to the user, e.g., BOSTON.	NA(A1)	YES	YES(C1)

(A1) DM-1 is indifferent to output device type.

(C1) If supported by the operating system.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO	NO	NO	NO	NO	NO

*U S ARMY SYSTEM DEVELOPED BY CDC

4.300 Sorting Capability

An important feature of output presentation is the capability to sort data into any desired order prior to, or simultaneously with, the actual output process. There are three main areas of interest here:

(a) optional features of a sort capability

(b) non-optional features of a sort capability which affect the sort environment and the efficiency of the sort process

(c) limitations on the sort capability.

4.310 General Characteristics

These are the characteristics which a user can control when the sort is used. They are:

(a) number and size of sort keys

(b) order of sort (ascending, descending, etc.)

4.320 Type of Sort

These are the technical sort capabilities not usually under the control of the user. They include:

(a) sort technique--whether records of a file are physically sorted, or whether a sorted index to the file is constructed instead

(b) auxiliary storage--what secondary storage devices, if any, are used in the intermediate stages of sorting

(c) system interface--whether the sort capability is part of the DMS, part of the associated operating system, or is independent of other software.

4.330 Sort Limitations

These are quantitative limitations on the data to be sorted and on the hardware resources which are required for the sorting. They include:

(a) the maximum sizes of data files which can be sorted at one time, the maximum size of records within a file, and the minimum number of records which must be presorted into standard size blocks

(b) the minimum amount of primary and secondary storage which must be available for the sort.

4.310 GENERAL CHARACTERISTICS	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
The general characteristics of the sort capability included in DMS's are:			
Number of Keys - The maximum number of different data elements that can be specified as sort keys.	16	25	(C1)
EXAMPLE: 3 sort keys - STATE, COUNTY, CITY 6 sort keys - STATE, COUNTY, CITY, WARD, PRECINCT, STREET			
Size of Keys - The maximum number of characters that can be used as sort keys.	254	63	(C1)
Order of Sort - The orders in which the information may be sorted. The basic order is that of the collating sequence of the system hardware.			
- Ascending	AV(F)	YES	(C1)
- Descending	AV(F)	YES	(C1)
- Ascending/Descending (if the sort program provides for different sort orders based on different keys)	AV(F)	YES	(C1)
EXAMPLE: SORT STATE ASCENDING AND CITY DESCENDING Output State — City ALABAMA — MONTGOMERY, MOBILE, BIRMINGHAM ARKANSAS — PINE BLUFF, LITTLE ROCK, FORT SMITH			

(C1) OS/360 sort.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NTL	9	5	99	10	9	10
NTL	248 BY	256	256	2550BY	99CH (I1)	1K CH
YES	YES	YES	YES	YES(H1)	YES	YES
YES	YES	YES	YES	YES	YES	YES
YES	YES	YES	NO(G1)	YES	YES	

(G1) Available in Phase II.

(H1) Also an ascending sort with elimination of duplicate data element values.

(I1) In a single report/query.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.310 GENERAL CHARACTERISTICS (Continued)	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
- User-Specified - Whether the user can specify a collating sequence that does not conform to the basic hardware collating sequence, such as A, C, B, D,...	NO	NO	(C1)
Note: Collating sequence is the generic term for the order of the computer character complement.			
4.320 TYPE OF SORT			
Sorts can be typed according to the amount of data transferred based on a key, the devices used during sorting, and whether they are provided as part of the operating system.			
Record - Entire data records are ordered based on the sort key.	AV(F)	YES	(C1)
Key - Data fields within records are extracted and ordered based on the sort key, and pointers to the data records are maintained.	AV(F)	YES	(C1)
Disc - Disc type memory is used as an auxiliary storage during sorting.	YES	YES	(C1)
Tape - Tapes are used for auxiliary storage during sorting.	NO	NO	(C1)
Part of Operating System - The DMS uses a sort program that is included in the hardware vendor's operating system, e.g., a DMS implemented on the IBM System 360, which uses the System 360 sort program.	YES	YES	YES

(C1) OS/360 sort.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NO	NO	NO	NO	NO	YES
YES	YES	NO	NO	NO	YES	NO
NO	YES	YES	YES	YES	NO	YES
YES	YES	YES	YES(G1)	NO	NO	YES
NO	YES	YES	YES(G1)	NO	YES	YES
NO	OS SORT	YES	YES	NO	NO	YES

(G1) Normally sorted on disc; tape sort when capacity is exceeded.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.330 SORT LIMITATIONS	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
The sort capability of a DMS may be limited by:			
<u>File Size</u> - The physical size of the entire data file.	(A1)	NTL	(C1)
<u>Tape Units</u> - The minimum number of tape units required by the sort program.	(A1)	NO	(C1)
<u>Record Size</u> - The maximum size of a data record.	(A1)	8000 CH	(C1)
<u>Memory</u> - The minimum amount of core storage required for the sort program.	(A1)	6K	(C1)
<u>Presort</u> - The input data to the sort program must be sorted by some larger grouping to provide workable-sized groups for the sort program.	(A1)	NO	(C1)

(A1) Limitations are those of the operating system.

(C1) OS/360 sort.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NTL(E1)	1 VOL	(G1)	NTL	YES(I1)	NTL
NO	NONE(E2)	NO	(G1)	NA	3	NO(J1)
NO	NO	YES	(G1)	NTL	NA	8K CH
NO	NO	NO	(G1)	NA	8K words	8K
NO	NO	NO	(G1)	NA	NO	NO

(E1) Only limitation is IBM Sort limits.
(E2) All input and output may be on any device supported by OS/360.

(G1) OS/360 sort.

(I1) Limited to one reel of input.

(J1) Tape Sort, four tape units.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.400 Data Reduction

Output functions are used to perform data reduction or summarization of the retrieved data. The resulting functional values will be provided to the user as part of the output presentation.

4.410 Statistical Functions

These are the basic distribution parameters of statistical theory.

4.420 Tallies

These are counts: the number of times that something occurs in the set of data which has been retrieved for output presentation. They include the number of instances of a data element and the number of occurrences of particular or unique data-element values.

4.430 Summarization

Tallies can be computed for an entire data file, or for portions of a data file. Summaries are sums or partial sums of the tallies which represent only a part of a data file.

4.410 STATISTICAL FUNCTIONS	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Maximum/Minimum	AV(F)	NO(B1)	NF
Mean	AV(F)	NO(B1)	YES
Median	AV(F)	NO(B1)	NF
Mode	AV(F)	NO(B1)	NF
Standard Deviation	AV(F)	NO(B1)	NF
Percent Total	AV(F)	NO(B1)	YES
4.420 TALLIES			
Count Unique Values - The ability of the system to count only the unique occurrences of data element value. EXAMPLE: For (SPRINGFIELD, ILL; SPRINGFIELD, MASS; SPRINGFIELD, VA) the city name is counted only once.	NO	NO(B1)	YES
Count All Instances			
- Data Element - The number of times a data element is found during the retrieval. EXAMPLE: QUERY: COUNT CITIES IN THE UNITED STATES OUTPUT: COUNT EQUALS 7500	AV(F)	NO(B1)	YES
- Data Element Value - The occurrences of a particular data-element value. EXAMPLE: QUERY: COUNT CITIES WITH POPU GREATER OR EQUAL TO 500,000 OUTPUT: NUMBER OF CITIES WITH POPULATION GREATER THAN 500,000 IS 100.	AV(F)	NO(B1)	YES

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	YES	YES	YES	YES	NO	NO
NO	YES	NO(F1)	YES	YES	NO	YES
NO	NO	NO(F1)	YES	NO	NO	YES
NO	NO	NO(F1)	YES	NO	NO	YES
NO	NO	NO(F1)	YES	YES	NO	NO
NO	YES	NO(F1)	YES	NO	NO	YES
NO	YES(E1)	YES	YES	YES	NO	YES
NO	YES	YES	YES	YES	YES	YES
NO	YES(E1)	YES	YES	YES	YES	YES

(D1) Available through COBOL.

(E1) By breaking on the data element.

(F1) Available through subroutine linkage.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

4.430 SUMMARIZATION	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
This indicates the ability of the system to provide summaries of data-element values retrieved.			
Data Elements - The maximum number of running totals that may be kept at any level in the structure of the file.	AV(F)	NTL(B1)	NTL
Data-Element Totals - The maximum number of different data element totals provided for by the system at any one time.	AV(F)	NTL(B1)	NTL
Subtotals - The maximum number of subtotals within a total.	AV(F)	NTL(B1)	NTL

(B1) Available through COBOL procedure.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	NTL	(F1)	(G1)	NTL	NA	NF(J1)
NO	NTL	(F1)	(G1)	NTL	NTL(I1)	NF
NO	NTL	(F1)	(G1)	NTL	5	NF

(F1) NTL (function of the execute module size of the generated procedure in automatic mode. Limited to 99 of each in specific mode (i. e., count 1 to count 99: V(1). . . V(99)).

(G1) Limited by size of core.

(I1) Limited by physical record length.

(J1) A capability to summarize across the rows and columns of a matrix is also planned.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

5.000 DATA ACCESS METHODS

This section describes the capabilities that the system provides for the user to access the data files.

5.100 On-Line Capabilities

The on-line capabilities of the system permit the user to interact with the system directly through remote consoles. They are characterized by the amount of traffic the system can handle, the ways by which the user interacts with the system and the ability of the system to establish priorities.

5.110 Traffic Volume

The number of on-line users which a computer can handle is always limited by its internal operating speed and by its primary and secondary storage capacity. Generally speaking, more on-line terminals can be physically connected to a computer than can be logically connected at any given time, and not all of those which are logically connected can be processed by the computer (i.e., not be in a wait state) simultaneously.

5.120 Man/Machine Interaction

This topic deals with the type of requests which the user can make and with the kind of responses the DMS can provide. It includes the subjects of extended dialog, tutorials, query correction, control of cross-connected terminals, query confirmation, and creation and use of prestored query procedures.

5.130 Priority Scheme Logic

This topic deals with the techniques used internally to determine the order in which system users are served by the system.

5.110 TRAFFIC VOLUME	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
This indicates the limits on the amount of traffic that the system can handle.			
Consoles/Terminals - The maximum number of on-line consoles or terminals that can be connected to the system.	NA(A1)	NF	NF
Active On-Line - The maximum number of consoles that may be active at any given time. This is not necessarily the same as the number that can be connected. For example, it may be possible to connect 50 terminals, but have only 20 active consoles at any given time.	NA(A1)	NF	NF
Simultaneous Users - This indicates the maximum number of on-line users who may have jobs being processed. This is not necessarily the same as the number of active on-line users. For example, because of hardware configuration restrictions (e. g., 5 fixed partitions of core in a multi-programming environment) it may be possible for only 5 of 20 active users to interact with the system simultaneously.	NA(A1)	2	NF

(A1) Entirely a function of host operating system.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	(E1)	LIM(O)	LIM(O)	10(H1)	NA	45(J1)
YES(D1)	(E1)	LIM(O)	LIM(O)	10(H1)	NA	45(J1)
YES(D1)	(E1)	LIM(O)	LIM(O)	10(H1)	NA	45(J1)

(D1) Available through TERMINAL SUPERVISOR.

(E1) On-line capabilities are not presently available.

(H1) Not a TDMS limit. The limit is a function of the operating system (ADEPT 50) as implementable on the IBM 360/50H. Other operating system implementations may provide for more terminals.

(J1) Capability available to add more terminals.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

5.120 MAN/MACHINE INTERACTION	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
Pre-stored Queries - Whether the user can request the execution of a pre-stored query directly from a remote console. The user must have the capability of selecting the specific query to be executed and be able to supply all necessary parameters from the console.	NF	AV(O)	YES
Query Composition - Whether the user can formulate and execute ad hoc retrieval requests directly from a remote console.	AV(F)	AV(O)	YES
Query Confirmation - Whether the system confirms receipt of user requests.	NF	NO	NF
Conversational Mode - Whether the user can engage in a dialogue with the system. This is usually a question/answer mode in which the user responds to system-provided questions or options in order to execute his request.	NF	YES	
- Always - The system provides no other method for user retrieval of data.	NF	NO	
- Optional - The user can query the system by other methods.	YES	YES	
Interactive Mode - Whether the system provides "walk-through" aids to the inexperienced user to help him formulate meaningful requests to the DMS.	AV(F)	NO	NF
Specify Output Station - Whether the user can direct output to a device other than his remote console.	NF	YES	NF
Erase - Whether the system provides for the correction of typographical errors on a text-word basis. This may be described as a "pseudo-backspace" function.	AV(F)	YES	NF

(D1) Available through operating system.
(D2) Available through TERMINAL SUPERVISOR.
(E1) On-line capabilities are not presently available.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NO(E1)	YES	NO(G1)	YES	NA	YES
NA	NO(E1)	YES	YES	YES	NA	YES
YES(D2)	NO(E1)	YES	YES	YES	NA	YES
NO	NO	NO	NO	YES	NA	NA
NA	NA	NA	NA	NO	NA	NA
NA	NA	NA	NA	YES	NA	NA
NA	NO(E1)	NO	NO(G2)	YES	NA	NO
YES(D2)	NO(E1)	YES(F1)	NO	YES	NA	YES
YES(D2)	NO(E1)	YES(F2)	NO(G3)	YES	NA	YES

(F1) Limited to off-line printer.
(F2) Function of user-written QTAM message control program.

(G1) Available in Phase II.
(G2) Limited to error diagnostics.
(G3) Must repeat a whole line.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

5.130 PRIORITY SCHEME LOGIC	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Priorities by			
- Type of user (e.g., military or civilian).	(A1)	(B1)	NF
- Task size (e.g., 3-minute run or 8-hour production run.	(A1)	NO	NF
- Terminal ID (e.g., the president's office or the 13th vice-president's office).	(A1)	(B1)	NF
- System status (e.g., if the system is busy, only urgent requests are acknowledged).	(A1)	(B1)	NF
Fixed Algorithm - A fixed algorithm implies that the system must be regenerated in order to change the priority algorithm; for example, it cannot be changed simply by reading a control card or typing in a special message.	(A1)	NO	NF
Dedicated System - In a dedicated system, the entir configuration is dedicated to DMS tasks. No other tasks can be processed at the same time. This does not imply, however, that the DMS task cannot be interrupted for one of a higher priority.	(A1)	NO	NO
Background Processing - Tasks processed in the background are generally long-duration batch-processing tasks. On-line tasks can compete with background tasks in two ways:			
- Lower Priority, Always Active - The background task assumes a lower priority than the on-line task; the on-line task can therefore usurp a portion of core.	(A1)	YES	NF

(A1) Entirely an operating system function.

(B1) Determined by operating system.

(D1) Available through operating system.

(E1) On-line capabilities are not presently available.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	(E1)	(F1)	NO	LIM(O)	NA	YES(J1)
NA	(E1)	(F1)	NO	LIM(O)	NA	YES
NA	(E1)	(F1)	NO(G1)	LIM(O)	NA	NO
YES(D1)	(E1)	(F1)	NO	LIM(O)	NA	NO
YES	YES(E1)	(F1)	YES	LIM(O)	NA	YES
NO	NO(E1)	NO	NO	LIM(O)	NA	NO
YES(D1)	(E1)	NA(F1)	YES	LIM(O)	NA	YES

(F1) Terminal service is based on QTAM priorities. GIS compiling sequence is based on QUEUE sequence and GIS can change dispatching priority. The Execute Module is scheduled using the OS/360 priority scheme (0-13).

(G1) Available in Phase II.

(J1) Priorities are: (1) immediate--90 seconds; (2) deferred--6 hours; (3) overnight--24 hours; (4) override--as soon as possible.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

5.130 PRIORITY SCHEME LOGIC (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- Low Priority, Rolled In and Out - The background task again assumes a lower priority than the on-line task; the on-line task in this case usurps the total core resource and the background task is checkpointed and written onto auxiliary storage.	(A1)	NO	NF

(A1) Entirely an operating system function.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	(E1)	NA(F1)	NO	LIM(O)	NA	AV(O) 1 QT 69

(E1) On-line capabilities are not presently available.

(F1) Terminal service is based on QTAM priorities. GIS compiling sequence is based on QUEUE sequence and GIS can change dispatching priority. The Execute Module is scheduled using the OS/360 priority scheme (0-13).

*U.S. ARMY SYSTEM DEVELOPED BY CDC

5.200 Security

This section indicates the types and levels of security that are provided for by the system in order to protect the information contained in the data base from unauthorized access. The following two major topics are considered:

(a) the security categories themselves

(b) the application of security categories to data-file protection.

5.210 GENERAL CAPABILITIES	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
This indicates the general levels of security provided by the system.			
Number of Categories - The maximum number of security levels the system provides for.	16	NA	NF
Cumulative Data Element/Group Access Analysis - Whether the system can specify higher classification for an entire report than the classification of its components; for example, each page of a printout may be classified as CONFIDENTIAL, but the total printout may be SECRET.	NO	NA	NF
Read and Write Protect - Access is restricted based on the type of access (read or write) coupled with security categories to protect data at various levels.			
- Read Protect - Prevents unauthorized retrieval of data at the following levels:			
● File	YES	AV(O)	YES
● Entry	YES	AV(O)	YES
● Group	YES	AV(O)	YES
● Data Element	YES	AV(O)	YES
- Write Protect - Prevents unauthorized modification of data at the following levels:			
● File	YES	YES	YES
● Entry	YES	YES	YES
● Group	YES	YES	YES
● Data Element	YES	YES	YES

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NA	NA	128	128	LIM(O)	5	3(J1)
NA	NO	NO	NO	LIM(O)	NO	NO
NA	(E1)	YES	YES	LIM(O)	NO	YES
NA	NO	NO	NO	NO	NO	YES
NA	NO	NO	NO	NO	NO	YES
NA	NO	YES	NO	NO	NO	YES
NA	NO	YES	NO	LIM(O)	NO	YES
NA	NO	NO	NO	NO	NO	YES
NA	NO	NO	NO	NO	NO	YES
NA	NO	YES	NO	NO	NO	YES

(E1) Files can be multiply defined with a particular field available to one definition and not to another. The security control will be as tight as the user's control of his operation. An undefined field is not available to any user without own-code intervention. No listings are supplied with the system.

(J1) CONFIDENTIAL, SECRET, and PRIVILEGED.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

6.000 GENERAL INTERFACE CAPABILITY

The functions provided by an operating system include task scheduling, resource allocation (core, devices and files), I/O control, monitoring, restart and recovery, and operator liaison. Utility-type software such as loaders, assemblers, compilers, and debugging aids may also be provided by the operating system.

Two types of operating system are identified:

(a) Vendor-supplied operating systems--i.e., those supplied by hardware manufacturers

(b) Independently-developed operating systems--i.e., those supplied by independent software developers.

This section indicates the type of operating system under which the DMS operates and the relationship of the DMS to the major components of this type of operating system.

6.100 General-Purpose Operating System

This section indicates the specific type of operating system under which the DMS operates and identifies those major components of the operating system which are used by the DMS.

6.110 Vendor-Supplied Operating System

This section tells whether the DMS uses a vendor-supplied operating system and, if so, identifies those major components of the vendor-supplied operating system which are used by the DMS.

6.120 Independently-Developed Operating System

This section tells whether the DMS uses an independently-developed operating system and, if so, identifies those major components of the independently-developed operating system which are used by the DMS.

	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
6.110 VENDOR-SUPPLIED OPERATING SYSTEM			
This indicates the extent to which the DMS uses facilities provided by a hardware-vendor-supplied operating system.			
<u>IOCS/Data Management</u> - Whether the DMS uses the input/output facilities of the operating system to read and write data.	YES	YES	YES
<u>Monitor</u> - Whether the DMS functions under the operating system monitor.	YES	YES	YES
<u>Scheduler</u> - Whether the DMS uses the scheduler of the operating system to order its jobs and tasks.	YES	YES	YES
<u>Language Processors</u> - Whether the DMS uses the language processors (i.e., FORTRAN, COBOL, etc.) supplied by the operating system.	YES	YES	YES
<u>Data Files</u> - Whether the DMS uses the data conventions of the operating system in handling data files.	YES	YES	YES
6.120 INDEPENDENTLY-DEVELOPED OPERATING SYSTEM			
This indicates the extent to which the DMS uses the facilities provided by an independently developed operating system.			
<u>IOCS/Data Management</u> (see 6.110)	NA	NO	NA
<u>Monitor</u> (see 6.110)	NA	NO	NA
<u>Scheduler</u> (see 6.110)	NA	NO	NA
<u>Language Processors</u> (see 6.110)	NA	NO	NA
<u>Data Files</u> (see 6.110)	NA	NO	NA

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES	YES	YES	YES	NO	NA	YES(J1)
YES	YES	YES	YES	NO	YES	YES
YES	YES	YES	YES	NO	NO	YES
YES	NO	YES	YES	NO	NO	YES
YES	YES	YES	YES	NO	NO	YES
NA	NA	NA	NO	YES	NA	NO
NA	NA	NA	NO	YES	NA	NO
NA	NA	NA	NO	YES	NA	NO
NA	NA	NA	NA	(H1)	NA	NO
NA	NA	NA	NO	(H1)	NA	NO

(H1) Assembly language and JOVIAL.

(J1) Master Operating System.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

6.200 Capability to Use Vendor's Software If Using Independently-Developed Operating System

This indicates the ability of a DMS operating under an independently-developed operating system to interface also with the vendor-supplied operating system. Two aspects are considered:

(a) what vendor's software capabilities can be used

(b) how these vendor's software capabilities can be used.

6.210 General Interface

This section specifies the features of the vendor-supplied software system which the DMS can use.

6.220 Formal Interface

This section describes the software mechanisms by means of which the DMS can call features of the vendor-supplied operating system for execution.

6.210 GENERAL INTERFACE	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Language Processors - FORTRAN, COBOL, etc.	NA	NA	NA
Applications Software - Report generators (RPG's), sort routines, text editors.	NA	NA	NA
Capability to Process Vendor's Software-Generated Files - Whether the DMS can process a data file generated under the vendor operating system.	NA	NA	NA
Capability to Load and Execute Programs Compiled Under Vendor's Software - Whether the DMS can load and execute an object program (i.e., FORTRAN or COBOL) compiled by the vendor's software.	NA	NA	NA
6.220 FORMAL INTERFACE			
Special Procedures - Whether a special interface exists to call the vendor's operating system.	NA	NA	NA
Standard Machine Bootstrap - Whether the vendor's operating system is loaded by a standard system startup procedure.	NA	NA	NA

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NA	NA	NA	NA	NO	NA	YES
NA	NA	NA	NA	NO	NA	NO
NA	NA	NA	NA	NF	NA	YES
NA	NA	NA	NA	NF	NA	YES
NA	NA	NA	NA	NO	NA	YES
NA	NA	NA	NA	YES	NA	YES

*U.S. ARMY SYSTEM DEVELOPED BY CDC

6.300 Dedicated Data Management Using a Modified Vendor's Operating System

This section indicates the extent, if any, to which the vendor's operating system components are modified to provide special facilities for the DMS when the operating system under which the DMS operates is a vendor-supplied operating system.

6.310 Special Functions

This section indicates which major components of the operating system have been specially modified for use by the DMS.

6.320 Vendor's Functions

This section indicates which major components of the operating system have not been modified at all for use by the DMS.

6.330 Formal Interface

This section describes the software mechanisms by means of which the DMS can call the modified vendor-supplied operating system.

6.310 SPECIAL FUNCTIONS	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
IOCS/Data Management (see 6.110)	NA	NA	NA
Monitor (see 6.110)	NA	NA	NA
Scheduler (see 6.110)	NA	NA	NA
Language Processor (see 6.110)	NA	NA	NA
6.320 VENDOR'S FUNCTIONS			
IOCS/Data Management (see 6.110)	NA	NA	NA
Monitor (see 6.110)	NA	NA	NA
Scheduler (see 6.110)	NA	NA	NA
Language Processor (see 6.110)	NA	NA	NA
330 FORMAL INTERFACE			
Special Procedures (see 6.220)	NA	NA	NA
Standard Machine Bootstrap (see 6.220)	NA	NA	NA

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NA	NA	NA	NA	NA	NA	NO
NA	NA	NA	NA	NA	NA	NO
NA	NA	NA	NA	NA	NA	NO
NA	NA	NA	NA	NA	NA	NO
NA	NA	NA	NA	NA	NA	YES
NA	NA	NA	NA	NA	NA	YES
NA	NA	NA	NA	NA	NA	YES
NA	NA	NA	NA	NA	NA	YES
NA	NA	NA	NA	NA	NA	YES
NA	NA	NA	NA	NA	NA	YES

*U.S. ARMY SYSTEM DEVELOPED BY CDC

7.000 SYSTEM STATISTICS

This section describes the capabilities in the DMS for generating and recording event, usage, and error statistics and times.

7.100 Event Recording

These are system-initiated tallies or counts of events, as well as lists and times of events and of their components.

7.110 Types of Events Recorded

These fall into three main categories:

(a) tallies of events
(b) total times for events
(c) logs for each job processed.

7.120 Recording Control Capability

This indicates the capability that the user has to specify the events to be recorded: predefined, conditional, dynamically selected.

7.130 Error Recording

The types of errors which can be detected and recorded:

(a) DMS errors
(b) operating system errors.

7.110 TYPES OF EVENTS RECORDED	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
System Tallies			
- Data Elements Retrieved - A count of the number of data elements retrieved.	AV(F)	NO	NF
- System Module Executed - A list of which system modules have been executed and how often.	NO(A1)	YES	NF
- Device Usage - Tallies of the number of consoles in use and the number of disc seeks issued.	NO(A1)	YES	NF
- Input Transactions - A tally of the number of transactions applied to a file and of those transactions which were rejected.	AV(F)	NO	YES
System Event Times			
- Job Execution - Total time for processing a specific job.	NO(A1)	YES	(C1)
- System Module Execution - Total processing time for each system module used.	NO(A1)	YES	(C1)
- Data Access - Total time required to search the file.	NO(A1)	NO	(C1)
System Logs			
- Standard Job Accounting - Such information could include, time on and time off the computer, programmer identification, normal or error termination condition, storage and device usage, system usage, etc., for each job.	NO(A1)	YES	YES

(A1) Operating system responsibilities.

(C1) Standard OS accounting.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	YES	YES	YES(G1)	NO	YES	NO
NO	NO	YES	YES	NO	YES	NO
NO	NO	NO(F1)	NO	NO	NA	NO
NO(D1)	YES	YES	YES	NO	YES	YES
YES(D2)	YES(E1)	AV(O)	AV(O)	LIM(O)	NO	YES
NO	NO	NO	NO	LIM(O)	NO	NO
NO	NO	AV(O)	AV(O)	LIM(O)	NO	NO
YES(D2)	NF	YES	AV(O)	LIM(O)	NO	YES

(D1) Available through COBOL.
(D2) Available through cperating system.
(E1) Standard OS accounting.
(F1) ISAM overflow chain usage statistics are available.
(G1) The number of records that qualified.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

7.110 TYPES OF EVENTS RECORDED (Continued)	AUERBACH	BURROUGHS	CSC
	DM-1	FORGE	COGENT III
- Operator Intervention - This log contains the history of operator interventions, i.e., when they occurred, and the reason.	NO(A1)	YES	YES
- Jobs Completed - This log indicates which jobs have terminated successfully	NO(A1)	YES	YES
7.120 RECORDING CONTROL CAPABILITY			
User Control Capability - Whether the user can specify the types of events that are to be recorded.			
- Selected Events - Whether the user can specify which events he wants to record	NO(A1)	NO	YES
- Predefined System Events Only - Whether the user has control over the events that are recorded.	NO(A1)	YES	NO
- Conditional Recording - Whether the user can conditionally record specific events.	NO(A1)	NO	YES
● Number of Levels - The number of categories that can be conditionally recorded by establishing appropriate parameters at system execution time.		2	NF
- All Defined Events - Once an event is selected, it is always recorded.	NO(A1)	YES	NO

(A1) Operating system responsibilities.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES(D1)	NF	NO	AV(O)	NO	NO	YES
YES(D1)	NF	YES	AV(O)	NO	NO	YES
NO	NO	YES(F1)	NO(G1)	NO	NO	NO
NO	NO	YES	NO	NO	NO	YES
NO	NO	YES	NO(G2)	NO	NO	YES
NO	NA	(F2)	NA	NA	NA	1
NO	YES	NO	NO(G2)	NO	NO	YES

(D1) Available through operating system.

(F1) 76 recording events (e.g., source file out of sequence, field failed validation check).

(F2) These 76 events are grouped into 21 categories, each of which can have level-of-importance and default options.

(G1) Limited to capabilities in OS/360.

(G2) At message level in OS/360.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

7.130 ERROR RECORDING	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
System Detected Errors - Whether the DMS can detect the following types of errors.	YES	NO	YES
- Data Element Values - These errors include invalid characters, incorrect number of characters, invalid values, etc.	YES	YES	YES
- Transaction Format - These errors include invalid field lengths, incorrect sequence of values, invalid transaction codes, etc.	YES	NO	YES
- Procedural Statements - These errors are those which involve the syntax or the punctuation of the procedural statements.			
OS Detected Errors - Whether the DMS can record errors by the operating system.			
- Equipment Malfunctions - Hardware generated errors.	NO	YES	YES
- Task or Job Specification Errors - These are errors in requesting the execution of a task or a job, e.g., incorrectly naming the program to be executed, incorrect device specification, etc.	NO	YES	YES

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
NO	YES	YES	YES	YES	YES	YES
NO	YES	YES	YES	YES	YES	YES
YES(D1)	YES	YES	YES	YES	YES	YES
YES(D2)	YES	YES	AV(O)	LIM(O)	YES	YES
YES(D3)	YES	YES	AV(O)	LIM(O)	YES	YES

(D1) Available through IDS and COBOL.
(D2) Available through FILE MANAGER.
(D3) Available through operating system.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

8.000 SYSTEM AVAILABILITY

This section identifies the hardware and software required to operate the DMS and the dates when the system and its documentation will be available.

8.100 Configuration and Availability

This section covers the following topics:

(a) minimum hardware requirements
(b) minimum software requirements
(c) availability of hardware and software.

8.110 Minimum Hardware Configuration

This section specifies the hardware components which are necessary for the operation of the DMS. It includes details of the central computer itself as well as of the secondary storage devices and remote consoles.

8.120 Basic Software Requirements

This section specifies the software components which are necessary for the operation of the DMS. It includes details of the operating system required to run the DMS, as well as any additional required software which is neither contained within the DMS itself nor part of the available operating system.

8.130 Availability

These are the dates when the hardware will be available for the implementation of the DMS and the dates when an initial and final DMS capability will be available. Initial capability assumes minimal retrieval and maintenance functions are available. Final capability includes all capability and functions identified in the survey.

8.110 MINIMUM HARDWARE CONFIGURATION	AUERBACH	BURROUGHS	CSC
	DM-I	FORGE	COGENT III
Computer Type (Mainframe)	360/50	B5500	360
Core Size	128K	24K(B1)	256K
Card Reader	1	1	1
Printer	1	2	1
Tape Drives	2	2	NF
Disc File	1	1	1
Consoles	1	1	NF
8.120 BASIC SOFTWARE REQUIREMENTS			
Operating System	OS 360	MCP	OS/MVT
Special Software	NO	COBOL	NONE
8.130 AVAILABILITY			
Hardware	NOW	NOW	NOW
Data Management System			
- Initial Capability	1 QT 69	NOW	4 QT 68
- Final Capability	4 QT 69	4 QT 67	1 QT 69

(B1) 48-bit words.

(D1) GE 200, 400, 600 series.

(E1) 32K DOS; 128K OS.

(F1) GIS (BASIC) will be available as an IBM Type-II package 30 June 1969. The system will require a minimum hardware configuration of a 360/40 with a G core (128K bytes). It will run under OS/360 in primary control mode (PCP).

(F2) GIS will be available as an IBM Type-II package 29 September 1969. The system will require a minimum hardware configuration of a 360/50 with an H core (512K bytes). It will run under OS/360 multiprogramming with a variable number of tasks (M VT).

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U.S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
(D1)	360/30	(F1)(F2)	(G1)	360/50	(I1)	CDC3300
32K	(E1)	(F1)(F2)	(G1)	256K	(I2)	48K
1	1	1	YES	1	1	1-405
1	1	1	YES	1	1	1-501
4	0	1	1(G2)	2(H1)	3	2-607
1 DSU	1-2311	3-2311	2311/14/02	1	NA	1-814
0	0	(F2)	1-2260 (G3)	2	NA	1
GECOS	OS. DOS	(F1)(F2)	OS	ADEPT	(I3)	MASTER
COBOL	NONE	NONE	NONE	NONE	NO	NONE
NOW	NOW	NOW	NOW	NOW	NOW	NOW
NOW	4 QT 67	(F1)	1 QT 68	2 QT 68	NOW	2 QT 68
NOW	1 QT 68	(F2)	3 QT 68	1 QT 69	NOW	NF

(G1) 360/40H for on-line capability. 360/40G-core for batch processing only.
(G2) Only 1 tape drive is required for tape input.
(G3) For on-line capability.
(H1) One 7-track unit; one 9-track unit.
(I1) SDS 910, 920, 925, 930 and 9300.
(I2) 8K on 900 series; 16K on 9300.
(I3) MONARCH for 900 series; MONITOR for 9300.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

8.200 Documentation

This section specifies the types of documents which are available to describe the DMS. There are two major classes of such documents that provide information concerning:

(a) how the system is constructed

(b) how to use the system.

8.210 System Specifications

These documents describe the internal workings of the DMS. They are divided into four subclasses of documents:

(a) system overview

(b) general system design

(c) detailed system design

(d) details of the implemented form of the system.

8.220 Operation Documentation

These documents describe the interface between the DMS and the outside world. There are two subclasses:

(a) the interface between the system and its users

(b) the interface between the system and the computer operators.

8.200 DOCUMENTATION	AUERBACH	BURROUGHS	CSC
8.210 SYSTEM SPECIFICATIONS	DM-1	FORGE	COGENT III
System Description (Overview)	YES	NO	YES
Conceptual System Description	YES	NO	YES
Detailed Design Specification	NF	NO	NF
Implementation Specification	NF	NO	YES
8.220 OPERATION DOCUMENTATION			
User			
- File Design Guide	(A1)	NO	YES
- Language Reference Manual	(A1)	YES(B1)	YES
- System Analysis Guide	(A1)	NO	YES
System Operation			
- Operator's Manual	(A1)	NO	YES
- System Maintenance	(A1)	NO	YES
- System Administrative Procedures	(A1)	NO	YES

(A1) Unless otherwise noted, final operating documentation is in various stages of planning or development and availability date is unknown.

(B1) FORGE Applications Systems Manual.

(E1) Proprietary.

(F1) Standard IBM Type II documentation supplied with system upon release--includes: Program Description Manual (Language Users Guide); Operations Manual; Systems Manual.

GE	INFORMATICS	IBM	NMCSSC	SDC	SDS	U S. ARMY*
IDS	MARK IV	GIS	NIPS	TDMS	MANAGE	RAPID
YES	NOW(E1)	NOW(F1)	YES	YES	YES	4 QT 67
YES	NOW(E1)	NOW(F1)	YES	YES	YES	4 QT 67
NO	NOW(E1)	(F2)	Draft(G1)	YES	YES	4 QT 67
NO	NOW(E1)	(F2)	Draft(G1)	(H1)	YES	4 QT 67
NO	NOW	(F2)	11/1/67	NO	YES	1 QT 68
YES	NOW	(F2)	12/15/67(G2)	YES(H2)	YES	1 QT 68
NO	NOW	(F2)	12/15/67(G2)	NO	YES	NA
LIM(O)	4 QT 67	(F2)	NF	NA	YES	2 QT 68
NO	NO	(F2)	NF	NA	YES	2 QT 68
NO	NO	(F2)	NF	NA	YES	2 QT 68

(F2) Application Description Manuals--GIS(Basic) H20-0571, GIS H20-0574.

(G1) Not generally available.

(G2) 15 December 1967 draft; 15 February 1968 final version.

(H1) Proprietary.

(H2) Interim User's Guides now available. Complete User's Manuals in preparation.

*U.S. ARMY SYSTEM DEVELOPED BY CDC

BIBLIOGRAPHY

Auerbach Corporation, Data Manager-1, Design Documentation Report, October 1, 1966.

Auerbach Corporation, Data Structuring Study for Interference Prediction Analysis, January 17, 1967.

Auerbach Corporation, Reliability Central Automatic Data Processing Subsystem, Vol. II, Design Specification Report, August 1966.

Bleier,R. E., The Time-Shared Data Management System (TDMS) Language Specifications, SDC Document TM-3370/000/00, April 12, 1967.

Burroughs Corporation, B5500 COBOL Reference Manual, Form 1024247, August 1965.

Burroughs Corporation, B5500 FORGE Application Systems Manual, Form 1028941, March 1967.

Burroughs Corporation, B5500 Operation Manual, Form 1024916, 1966.

Dowkont, et al., A Methodology for Comparison of Generalized Data Management Systems. PEGS (Parametric Evaluation of Generalized Systems) ESD-TR-67-2, March 1967.

General Electric Company, GE-625/635 Integrated Data Store, No. CPB-1093, October 1966.

General Electric Company, GE-625/635 Sort/Merge Program, No. CPB-483.

General Electric Company, Integrated Data Store, A New Concept in Data Management, No. CPB-483.

General Electric Company, IDS/COBOL Reference Manual, No. CPB-1194.

General Electric Company, Introduction to Integrated Data Store, No. CPB-1048, April 1965.

Informatics, Inc., MARK IV, File Management System, Form SDO I, 5 M 867, 1967.

National Military Command System Support Center, Program Design Approach, System 360/50 Formatted File System, No. SPM 1-67, January 17, 1967.

Raucher, Virginia and H. S. Schwimmer, The Basic Language Specifications for TDMS, SDC Document TM-3370/001/00, April 10, 1967.

Raucher, Virginia, The Language Specifications for the Query Operation of TDMS, SDC Document TM-3370/004/00, April 12, 1967.

Raucher, Virginia, The Language Specifications for the Update Operations of TDMS, SDC Document TM-3370/005/00, June 28, 1967.

Scientific Data Systems, SDS Business Programming System, 1965.

Scientific Data Systems, SDS MANAGE Reference Manual, No. 99 10 46A, May 1966.

Vorhaus, Alfred H. and Robert D. Wills, The Time-Shared Data Management System: A New Approach to Data Management, SP-2747, February 13, 1967.

Williams, W. D. and P. R. Bartrom, COMPOSE/PRODUCE: A User-Oriented Report Generator Capability Within the SDC Time-Shared Data Management System, SDC Document SP-2634, February 8 1967.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* The MITRE Corporation 1820 Dolley Madison Boulevard McLean, Virginia 22101	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP N/A

3. REPORT TITLE

Data Management Systems Survey

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

N/A

5. AUTHOR(S) *(First name, middle initial, last name)*

James P. Fry	Walter P. Grabowsky	Herbert J. Sternick
Samuel Bramson	John Jeffries, Jr.	
David C. Fried	Suzanne B. Mahle	

6. REPORT DATE January 1969	7a. TOTAL NO. OF PAGES 210	7b. NO. OF REFS 22
8a. CONTRACT OR GRANT NO. F19628-68-C-0365 b. PROJECT NO. 33.4.B c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) MTP-329 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* None	

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited

11. SUPPLEMENTARY NOTES Data Management Series No. 2	12. SPONSORING MILITARY ACTIVITY Dep. Dir., NMCSTS Defense Communications Agency, South Court House Road, Arlington, Virginia 22204, Attn: Mr. Reynold Thomas, Jr., Code 912

13. ABSTRACT

This report presents the results of a survey of salient characteristics of a representative set of state-of-the-art data management systems. It is part of an effort to identify the state-of-the-art capabilities of data management systems for third-generation computer systems.

Section I of the report includes general descriptions of the systems surveyed and establishes the terminology for logical organization of data used in the survey.

Section II describes the capabilities surveyed and presents the survey results in tabular format.

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Data Management Systems						
Computers						
Data Structure						
Query Languages						
Information File Processing						
Retrieval						
Compilers						
Data Bases						
Data Management						
File Indexing						
Information Systems						
File Maintenance						
Data Validation						
POL						

UNCLASSIFIED
Security Classification